

UNIVERSIDADE FEDERAL DO PARANÁ

CARISE ELISANE SCHMIDT

PROBLEMAS DE LOCALIZAÇÃO E ROTEAMENTO DEPENDENTES DO TEMPO

CURITIBA PR

2019

**CARISE ELISANE SCHMIDT**

**PROBLEMAS DE LOCALIZAÇÃO E ROTEAMENTO DEPENDENTES DO TEMPO**

Tese apresentada ao Curso de Pós-Graduação em Métodos Numéricos em Engenharia, Área de Concentração em Programação Matemática, Setores de Tecnologia e Ciências Exatas, Universidade Federal do Paraná, como requisito parcial à obtenção do título de Doutor em Ciências.

Orientador: Prof. Dr. Arinei Carlos Lindbeck da Silva.

**CURITIBA PR**

**2019**

**Catálogo na Fonte: Sistema de Bibliotecas, UFPR  
Biblioteca de Ciência e Tecnologia**

**S349p**

**Schmidt, Carise Elisane**

**Problemas de localização e roteamento dependentes do tempo [recurso eletrônico] /Carise Elisane Schmidt. – Curitiba, 2019.**

**Tese – Universidade Federal do Paraná, Setor de Tecnologia, Programa de Pós-Graduação em Métodos Numéricos em Engenharia, 2019.**

**Orientador: Arinei Carlos Lindbeck da Silva.**

**1. Cadeia de logística integrada. 2. Logística. 3. Programação heurística. 4. Otimização combinatória. I. Universidade Federal do Paraná. II. Silva, Arinei Carlos Lindbeck da. III. Título.**

**CDD: 519.6**

**Bibliotecária: Vanusa Maciel CRB- 9/1928**

## TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em MÉTODOS NUMÉRICOS EM ENGENHARIA da Universidade Federal do Paraná foram convocados para realizar a arguição da tese de Doutorado de **CARISE ELISANE SCHMIDT** intitulada: **Problemas de Localização e Roteamento Dependentes do Tempo**, sob orientação do Prof. Dr. ARINEI CARLOS LINDBECK DA SILVA, que após terem inquirido a aluna e realizada a avaliação do trabalho, são de parecer pela sua aprovação no rito de defesa.

A outorga do título de doutor está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 17 de Dezembro de 2019.



ARINEI CARLOS LINDBECK DA SILVA  
Presidente da Banca Examinadora



SERGIO FERNANDO MAYERLE  
Avaliador Externo (UNIVERSIDADE FEDERAL DE SANTA CATARINA)



ADEMIR ALVES RIBEIRO  
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)



LEANDRO CALLEGARI COELHO  
Avaliador Externo (UNIVERSITÉ LAVAL)



LEANDRO MAGATAO  
Avaliador Externo (UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ)

*Aos meus pais, de quem recebi o  
maior e melhor presente de todos: a  
vida!*

## AGRADECIMENTOS

A Deus, por absolutamente tudo!

Aos meus pais, Maria e Waldemiro, e a minha irmã Clenise, pelo amor incondicional, incentivo e apoio que tornaram tudo mais fácil.

Ao meu orientador, Professor Arinei Carlos Lindbeck da Silva, pela paciência, dedicação, amizade e disponibilidade em todas as horas. Obrigada por aceitar o desafio desta pesquisa e confiar no meu trabalho. Agradeço por todos os momentos de discussão e por compartilhar, incansavelmente e sempre com entusiasmo, seus conhecimentos.

Aos Professores Leandro Callegari Coelho e Maryam Darvish, pela oportunidade de realização do estágio no CIRRELT (*Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport*), pela sugestão do tema de pesquisa, incentivo e receptividade recebidos, e pelo suporte técnico. Obrigada pela sempre pronta disponibilidade para discussão, inclusive à distância, e por toda experiência e conhecimento compartilhados.

À Universidade Federal do Paraná - UFPR, e ao Programa de Pós-Graduação em Métodos Numéricos em Engenharia - PPGMNE, pelo espaço aberto. Agradeço aos professores, pelos conhecimentos transmitidos, e aos secretários do programa, pelo suporte técnico.

Ao apoio técnico, sempre prontamente recebido, dos colegas Cléder, Alyson e Roberto, na execução do *solver* Gurobi e com os primeiros passos na linguagem de programação C++; e aos pesquisadores Hamza e Khaled, pelo suporte na geração das instâncias de teste.

A todos os amigos que, de longe ou de perto, ofereceram seu apoio e incentivo. Gratidão aos amigos que ganhei durante esse período, pelas horas de convívio e amizade que tornam essa etapa muito melhor.

Ao Instituto Federal de Santa Catarina - IFSC, pela oportunidade de afastamento integral das atividades.

Ao Grupo de Tecnologia Aplicada à Otimização - GTAO, pelos recursos disponibilizados.

Aos membros da banca examinadora, Prof. Leandro Callegari Coelho, Prof. Sergio Fernando Mayerle, Prof. Leandro Magatão, Prof. Ademir Alves Ribeiro e Prof<sup>a</sup>. Mariana Kleina, pelo aceite para participar da banca, pela avaliação do trabalho e pelas valiosas contribuições.

Muito obrigada!

## RESUMO

O problema de localização e roteamento é amplamente estudado na literatura. Sua popularidade se deve, principalmente, à importância da logística integrada e aos crescentes esforços para fornecer soluções eficientes para sua resolução. Ao integrar decisões de localização de instalações e roteamento de veículos, o problema visa minimizar o custo total a elas associado. Contudo, ao considerar o custo de roteirização apenas a partir da distância, assume-se uma simplificação que não traduz apropriadamente a realidade. Isso porque há mudanças no tráfego que alteram sensivelmente o tempo de viagem, fazendo com que o percurso mais curto nem sempre seja o mais rápido. Essa alegação deu origem ao problema de roteamento de veículos dependente do tempo e, apesar do interesse crescente por essa variante, a questão ainda apresenta lacunas na área de logística integrada. Diante disso, propõe-se integrar dois tópicos populares da pesquisa: o problema de roteamento de veículos dependente do tempo e o problema de localização e roteamento. O objetivo é definir, modelar e resolver problemas para os quais a velocidade não é considerada constante. O trabalho apresenta a primeira formulação matemática para o problema de localização e roteamento dependente do tempo, fortalecida por um conjunto de desigualdades válidas. Assume-se frota homogênea e limitada, e horário fixo de saída dos veículos para atendimento aos clientes. Busca-se definir o depósito com a melhor localização e as rotas que devem ser executadas para minimizar o tempo total de viagem. Propõe-se ainda um algoritmo metaheurístico, que combina heurísticas construtivas com um modelo de particionamento de conjuntos. Extensivos testes computacionais foram conduzidos para comparar as abordagens propostas, em um conjunto de instâncias gerado com base em dados reais de tráfego. Usando um *solver* comercial, o modelo foi capaz de fornecer soluções para instâncias com até 100 clientes e 15 intervalos de tempo, enquanto as desigualdades válidas melhoraram os limites inferiores. Os resultados evidenciam a importância da integração entre métodos heurísticos e exatos. Além disso, este trabalho contempla ainda uma extensão do problema, com dimensionamento de frota heterogênea. Nela, consideram-se depósitos capacitados, custos fixos de utilização de depósitos e veículos, frota heterogênea e não limitada, e flexibilidade no momento de saída dos veículos nos depósitos. Novamente, duas abordagens de resolução são propostas: uma formulação matemática com um conjunto de desigualdades válidas, e um algoritmo meta-heurístico de três fases baseado em busca evolucionária. Experimentos computacionais foram conduzidos para comparar essas abordagens. Os resultados reforçam a importância das desigualdades válidas na melhoria dos limites inferiores. Eles também mostram que a abordagem meta-heurística é capaz de reduzir consideravelmente o tempo de processamento e elevar a qualidade média das soluções.

**Palavras-chave:** Logística integrada. Roteamento de veículos dependente do tempo. Localização de instalações. Dimensionamento de frota heterogênea.

## **ABSTRACT**

Location-routing problem is a widely studied problem in the literature. Its popularity is mainly due to the importance of the integrated logistics and the increasing efforts to provide efficient solutions to these problems. It integrates facility location and vehicle routing decisions and as such, it minimizes the facility as well as the routing costs. However, assuming that the routing cost only depends on the distance traveled is an oversimplification which is inconsistent with the reality. In fact, changes in the traffic pattern significantly influence the travel time. Therefore, the shortest route is not always the fastest one. This argument has given rise to the emergence of the time-dependent vehicle routing problem. Despite the recent growing interest for this variant, there is a gap in the integrated logistics and models with this regard. To fulfill this lack of the literature, this thesis proposes to integrate two popular research areas: the time-dependent vehicle routing and the location-routing problems. The purpose of this thesis is to define, model, and solve integrated routing problems in which the speed is not considered constant over time. It presents the first mathematical formulation for the time-dependent location-routing problem, which is strengthened by a set of valid inequalities. A homogeneous and limited fleet is considered while the departure time from the depot to serve the customers is fixed. The objective is to minimize the total travel time while identifying the best location for a single depot and the routes that must be taken to serve customers. Moreover, a matheuristic algorithm is proposed, which combines constructive heuristics with a set partitioning model. Using a set of instances generated based on real traffic data, extensive computational experiments were conducted to assess the performance of proposed methods. With a commercial solver, the mathematical model was able to provide solutions for instances with up to 100 customers and 15 time intervals, obtaining improved lower bounds using valid inequalities. The results show the benefits of combining heuristic and exact methods. Furthermore, this research extends the problem to the case with heterogeneous fleet sizing. In this problem setting, capacitated depots, fixed costs of depots and vehicles, heterogeneous and non-limited fleet, and flexibility for the vehicle departure time from depots are considered. Once more, two different approaches to solve the problem are proposed: a mathematical formulation, with a set of valid inequalities, and a three-phase evolutionary search-based metaheuristic algorithm. Computational experiments were conducted on a set of instances to compare the performance of the developed methods. The results highlight the importance of valid inequalities to improve the lower bounds. They further indicate how the proposed metaheuristic considerably reduces the execution time and improves the quality of the obtained solutions.

**Keywords:** Integrated logistics. Time-dependent vehicle routing. Facility location. Fleet size and mix.



## LISTA DE TABELAS

3.1	NOTAÇÃO USADA NO PLR-DT . . . . .	29
3.2	CONJUNTOS DE INSTÂNCIAS E SEUS PARÂMETROS. . . . .	44
3.3	INFORMAÇÕES SOBRE TEMPO DE DESLOCAMENTO PARA INSTÂNCIAS COM 3 DEPÓSITOS. . . . .	44
3.4	RESULTADOS MÉDIOS DAS HEURÍSTICAS CONSTRUTIVAS PARA INSTÂNCIAS COM 1 DEPÓSITO. . . . .	45
3.5	RESULTADOS MÉDIOS DAS HEURÍSTICAS CONSTRUTIVAS PARA INSTÂNCIAS COM 3 DEPÓSITOS . . . . .	46
3.6	RESULTADOS MÉDIOS DAS HEURÍSTICAS CONSTRUTIVAS PARA INSTÂNCIAS COM 5 DEPÓSITOS . . . . .	46
3.7	RESULTADOS MÉDIOS DO PLR-DT COM DESIGUALDADES VÁLIDAS E SOLUÇÕES INICIAIS PARA INSTÂNCIAS COM 1 DEPÓSITO . . . . .	47
3.8	RESULTADOS MÉDIOS DO PLR-DT COM DESIGUALDADES VÁLIDAS E SOLUÇÕES INICIAIS PARA INSTÂNCIAS COM 3 DEPÓSITOS . . . . .	48
3.9	RESULTADOS MÉDIOS DO PLR-DT COM DESIGUALDADES VÁLIDAS E SOLUÇÕES INICIAIS PARA INSTÂNCIAS COM 5 DEPÓSITOS . . . . .	49
3.10	RESULTADOS MÉDIOS DO PLR-DT SEM SOLUÇÕES INICIAIS E COM DESIGUALDADES VÁLIDAS. . . . .	50
3.11	RESULTADOS MÉDIOS DO PLR-DT COM SOLUÇÕES INICIAIS E SEM DESIGUALDADES VÁLIDAS. . . . .	51
3.12	RESULTADOS MÉDIOS DA MATEURÍSTICA PARA INSTÂNCIAS COM 1 DEPÓSITO . . . . .	53
3.13	RESULTADOS MÉDIOS DA MATEURÍSTICA PARA INSTÂNCIAS COM 3 DEPÓSITOS . . . . .	54
3.14	RESULTADOS MÉDIOS DA MATEURÍSTICA PARA INSTÂNCIAS COM 5 DEPÓSITOS . . . . .	55
3.15	RESULTADOS MÉDIOS PARA INSTÂNCIAS COM 10 CLIENTES E 3 DEPÓSITOS: COMPARAÇÃO ENTRE O PLR-DT E O PRV-DT SEQUENCIAL	55
3.16	RESULTADOS MÉDIOS PARA INSTÂNCIAS COM 10 CLIENTES E 5 DEPÓSITOS: COMPARAÇÃO ENTRE O PLR-DT E O PRV-DT SEQUENCIAL	56

4.1	NOTAÇÃO USADA NO PLRDFH-DT . . . . .	61
4.2	CONJUNTOS DE INSTÂNCIAS PARA O PLRDFH-DT E SEUS PARÂMETROS . . . . .	81
4.3	RESULTADOS MÉDIOS DO PLRDFH-DT COM SOLUÇÕES INICIAIS . . .	83
4.4	RESULTADOS MÉDIOS DO PLRDFH-DT COM SOLUÇÕES INICIAIS E DESIGUALDADES VÁLIDAS. . . . .	84
4.5	RESULTADOS MÉDIOS DA ABORDAGEM META-HEURÍSTICA . . . . .	86
4.6	RESULTADOS MÉDIOS DA ABORDAGEM META-HEURÍSTICA NA RESOLUÇÃO DO PLR-DT PARA INSTÂNCIAS COM 3 DEPÓSITOS . . . . .	88
4.7	RESULTADOS MÉDIOS DA ABORDAGEM META-HEURÍSTICA NA RESOLUÇÃO DO PLR-DT PARA INSTÂNCIAS COM 5 DEPÓSITOS . . . . .	89

## LISTA DE ALGORITMOS

Algoritmo 1	Critério de parada para heurísticas construtivas . . . . .	35
Algoritmo 2	Rotas aleatórias . . . . .	36
Algoritmo 3	Replicação nos demais depósitos . . . . .	36
Algoritmo 4	Rotas aleatórias paralelas . . . . .	37
Algoritmo 5	$\nu$ - vizinhos mais próximos . . . . .	39
Algoritmo 6	$\nu$ - vizinhos mais próximos – paralela . . . . .	40
Algoritmo 7	Enumeração de até $\gamma$ clientes . . . . .	41
Algoritmo 8	Estrutura geral do algoritmo meta-heurístico . . . . .	69
Algoritmo 9	Melhoria intra-rota . . . . .	71
Algoritmo 10	Inicialização . . . . .	73
Algoritmo 11	Evolução . . . . .	77
Algoritmo 12	Intensificação . . . . .	79

## LISTA DE SIGLAS

ACO	<i>Ant Colony Optimization</i>
ALNS	<i>Adaptive Large Neighborhood Search</i>
ELS	<i>Evolutionary Local Search</i>
FIFO	<i>First In First Out</i>
FSM	<i>Flow Speed Model</i>
GTS	Granular Tabu Search
GRASP	<i>Greedy Randomized Adaptive Search Procedure</i>
ILP	<i>Instruction Level Parallelism</i>
LTM	<i>Link Travel Time Model</i>
PLF	Problema de Localização de Facilidades
PLR	Problema de Localização e Roteamento
PLR-DT	Problema de Localização e Roteamento Dependente do Tempo
PLRDFH-DT	Problema de Localização e Roteamento Dependente do Tempo com Dimensi- onamento de Frota Heterogênea
PRV	Problema de Roteamento de Veículos
PRV-DT	Problema de Roteamento de Veículos Dependente do Tempo
SA	<i>Simulated Annealing</i>
VND	<i>Variable Neighborhood Descent</i>
VNS	<i>Variable Neighborhood Search</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO . . . . .</b>	<b>13</b>
1.1	OBJETIVO GERAL . . . . .	14
1.2	OBJETIVOS ESPECÍFICOS . . . . .	14
1.3	INOVAÇÃO E CONTRIBUIÇÕES. . . . .	15
1.4	DELIMITAÇÕES DA PESQUISA . . . . .	15
1.5	ESTRUTURA DO TRABALHO . . . . .	16
<b>2</b>	<b>REVISÃO DE LITERATURA . . . . .</b>	<b>17</b>
2.1	O PROBLEMA DE LOCALIZAÇÃO E ROTEAMENTO. . . . .	17
2.1.1	Variações e extensões do problema . . . . .	19
2.1.2	Abordagens de resolução da literatura . . . . .	23
2.2	A DEPENDÊNCIA DO TEMPO EM PROBLEMAS DE ROTEAMENTO. . . . .	24
2.2.1	Considerações sobre a propriedade FIFO. . . . .	26
<b>3</b>	<b>PROBLEMA DE LOCALIZAÇÃO E ROTEAMENTO DEPENDENTE DO TEMPO. . . . .</b>	<b>27</b>
3.1	DESCRIÇÃO DO PROBLEMA E FORMULAÇÃO MATEMÁTICA . . . . .	27
3.1.1	Desigualdades válidas. . . . .	32
3.2	ALGORITMO MATEURÍSTICO . . . . .	34
3.2.1	Heurísticas construtivas. . . . .	34
3.2.2	Modelo de particionamento de conjuntos. . . . .	41
3.3	TESTES COMPUTACIONAIS. . . . .	42
3.3.1	Geração de instâncias . . . . .	43
3.4	RESULTADOS E DISCUSSÃO . . . . .	44
3.4.1	Resultados das heurísticas construtivas . . . . .	45
3.4.2	Resultados da formulação matemática . . . . .	47
3.4.3	Resultados da abordagem mateurística . . . . .	52
3.4.4	Comparação entre a formulação matemática e uma abordagem sequencial . . . . .	53
3.5	CONCLUSÕES . . . . .	57

<b>4</b>	<b>PROBLEMA DE LOCALIZAÇÃO E ROTEAMENTO DEPENDENTE DO TEMPO COM DIMENSIONAMENTO DE FROTA HETEROGÊNEA . . .</b>	<b>58</b>
4.1	DESCRIÇÃO DO PROBLEMA E FORMULAÇÃO MATEMÁTICA . . . . .	58
4.1.1	Desigualdades válidas. . . . .	65
4.2	ALGORITMO META-HEURÍSTICO . . . . .	67
4.2.1	Inicialização. . . . .	68
4.2.2	Evolução . . . . .	74
4.2.3	Intensificação . . . . .	76
4.3	TESTES COMPUTACIONAIS. . . . .	78
4.3.1	Instâncias de teste . . . . .	80
4.4	RESULTADOS E DISCUSSÃO . . . . .	81
4.4.1	Resultados da formulação matemática . . . . .	81
4.4.2	Resultados da abordagem meta-heurística . . . . .	85
4.4.3	Desempenho do algoritmo meta-heurístico na resolução do PLR-DT . . . . .	87
4.5	CONCLUSÕES . . . . .	89
<b>5</b>	<b>CONSIDERAÇÕES FINAIS . . . . .</b>	<b>90</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>93</b>
<b>6</b>	<b>APÊNDICE A . . . . .</b>	<b>99</b>

## 1 INTRODUÇÃO

O planejamento da cadeia de suprimentos é uma questão relevante para a maioria das empresas, uma vez que os custos associados com esse setor normalmente compõem uma fração significativa do seu orçamento (CAPELLE et al., 2019). A logística é uma das etapas que integra essa cadeia. Ela visa fornecer soluções eficientes em diferentes níveis da tomada de decisão. A rede de distribuição instalada ao final da cadeia representa uma parcela importante nessa organização, pois ela envolve diversos pequenos fluxos de produtos na direção de clientes finais ou de revendedores (PRODHON; PRINS, 2014).

A estreita relação entre desenvolvimento econômico e consumo logístico (BOWER-SOX; CALANTONE; RODRIGUES, 2003) torna, no entanto, o transporte de mercadorias especialmente desafiador quando este envolve áreas urbanas. Ao mesmo tempo que a logística e o transporte possibilitam o desenvolvimento econômico, eles são também, tradicionalmente, ineficientes em zonas urbanas (CLEOPHAS et al., 2019). Entre os principais fatores que contribuem para essa ineficiência está a ocorrência de congestionamento no tráfego.

Outro desafio do transporte de mercadorias, dentro e em torno de zonas urbanas, são os impactos sociais, ambientais e econômicos provocados nessas áreas (BEKTAŞ; CRAINIC; WOENSEL, 2017; NATARAJ et al., 2019). Isso já tem levado muitas cidades a investir na melhoria de seus sistemas de transporte (YUAN; YU, 2018). No entanto, na grande maioria dos casos, o foco principal nesses investimentos ainda é a minimização de custos (DEMIR; BEKTAŞ; LAPORTE, 2014). Dessa forma, projetar uma rede eficiente, que evite, por exemplo, o congestionamento de veículos, tem sido uma preocupação menos evidente.

No que tange às empresas, desconsiderar o congestionamento no tráfego pode acarretar inúmeros inconvenientes na sua logística de distribuição. Isso inclui chegadas tardias para atendimento de clientes (VEENSTRA; COELHO, 2017) e custos adicionais, como a necessidade de contratação de motoristas para cumprimento de prazos (MANCINI, 2017). Diferentes aspectos podem causar o congestionamento, mas alguns são previsíveis e ocorrem de forma sistemática em determinados horários e regiões, sendo possível considerá-los no planejamento.

Dentro desse contexto, duas decisões exercem um papel fundamental: a localização de instalações e a determinação de rotas para atender os clientes, a partir delas (KOÇ et al., 2016). E apesar de sua natureza distinta, essas duas decisões são interdependentes (PRINS et al., 2007; YU et al., 2010; PRODHON; PRINS, 2014), e negligenciar essa dependência pode ocasionar custos adicionais às empresas (SALHI; RAND, 1989).

Contudo, na literatura, as decisões de localização e distribuição foram estudadas separadamente durante muito tempo. Além disso, uma vez integrados, os modelos apresentam simplificações que nem sempre refletem a realidade. A maioria deles tem como objetivo a

minimização de custos, tendo a distância percorrida como base para o custo operacional de atendimento aos clientes.

Do ponto de vista estratégico, isso gera uma tendência a localizar instalações distantes dos centros urbanos, especialmente na periferia das cidades (KOÇ et al., 2016). No entanto, embora a diferença entre os custos associados às instalações se mantenha, o padrão de tráfego pode diferir consideravelmente entre as regiões onde elas estão situadas. Do ponto de vista operacional, a premissa de tempo de deslocamento constante é uma simplificação que, em geral, não representa corretamente a realidade. Isso porque, ao longo do dia, ocorrem mudanças no tráfego que alteram sensivelmente o tempo de deslocamento dos veículos (MANCINI, 2017), fazendo com que o trajeto mais curto nem sempre seja o mais rápido.

Focando nesse aspecto, o intuito deste trabalho é estender a classe de problemas que integra decisões de localização e de roteamento de veículos, para incluir problemas nos quais o tempo de deslocamento não é considerado constante. Com isso, busca-se preencher uma lacuna existente entre dois tópicos populares de pesquisa: o problema de localização e roteamento (PLR) e o problema de roteamento de veículos dependente do tempo (PRV-DT). A contribuição desta pesquisa reside na integração dessas duas áreas e na proposição de problemas de localização e roteamento dependentes do tempo.

## 1.1 OBJETIVO GERAL

O objetivo geral desta tese é propor, modelar e resolver, usando abordagens exatas e heurísticas, um novo problema de localização e roteamento que considera a dependência de tempo no deslocamento dos veículos; e também uma extensão desse problema, que inclui ainda o dimensionamento de frota heterogênea.

## 1.2 OBJETIVOS ESPECÍFICOS

O alcance do objetivo geral desta pesquisa compreende os seguintes objetivos específicos:

- Definir formalmente cada um dos problemas, especificando suas características e estabelecendo os parâmetros e variáveis envolvidas;
- Propor um modelo matemático de programação linear inteira mista e um conjunto de desigualdades válidas para cada um dos casos;
- Propor, para cada problema, uma abordagem de resolução heurística;
- Criar um conjunto de instâncias de teste que considere a dinâmica real de tráfego urbano;
- Demonstrar a eficiência e robustez dos métodos propostos através de extensivos testes computacionais;
- Contribuir com literatura da área, apresentando uma base para trabalhos futuros.



### 1.3 INOVAÇÃO E CONTRIBUIÇÕES

A inovação deste trabalho consiste na introdução de um novo problema de localização e roteamento que considera a dependência de tempo em seu aspecto operacional; e de uma extensão deste, que inclui também o dimensionamento de frota heterogênea. No primeiro caso, o objetivo é minimizar o tempo total de execução das rotas, considerando a abertura de um único depósito, horário fixo de saída dos veículos no depósito e frota limitada e homogênea. Na sequência, estende-se este problema para incluir depósitos capacitados, flexibilidade no horário de saída dos veículos no depósito, frota heterogênea e não limitada, e incidência de custos fixos sobre a abertura de depósitos, uso de veículos e no tempo de viagem, visando minimizar a soma total desses custos. Ao abordar problemas ainda não evidenciados na literatura, esta tese contribui teoricamente com o desenvolvimento da pesquisa na área em que está inserida.

Duas formulações matemáticas distintas, descritas em termos de programação linear inteira mista, são propostas, conferindo um caráter inédito ao trabalho. Cada uma é acompanhada de um conjunto de desigualdades válidas, que reforçam os limites inferiores desses modelos. Essas formulações podem servir como base para a incorporação de novas variáveis, a proposição e modelagem de novas extensões ou variantes do problema e aplicação em casos reais.

A complexidade combinatorial dos modelos apresentados motiva o desenvolvimento de abordagens heurísticas de solução. Dois algoritmos distintos são propostos. O primeiro, um algoritmo mateurístico<sup>1</sup>, engloba heurísticas construtivas de geração de rotas com um modelo exato de particionamento de conjuntos para tratar o problema de localização e roteamento dependente do tempo (PLR-DT). O segundo constituiu um algoritmo meta-heurístico baseado em busca evolucionária, que inclui procedimentos de diversificação, educação e intensificação para melhoria da qualidade das soluções. Ele foi proposto para tratar também a extensão do problema de localização e roteamento dependente do tempo, que inclui dimensionamento de frota heterogênea. Ambos contêm ideias já aplicadas na literatura, mas contemplam estratégias e procedimentos inéditos que tratam do caráter dinâmico associado com a inclusão da dependência temporal. As ideias lançadas aqui também podem servir como suporte para a proposição de novas abordagens de resolução em problemas com dependência de tempo.

### 1.4 DELIMITAÇÕES DA PESQUISA

A complexidade presente na proposição e resolução de um problema de logística integrada requer que algumas simplificações da realidade sejam feitas, como forma de gerar um modelo tratável. Esta pesquisa limita-se ao estudo de problemas determinísticos. Informações como demanda de clientes, tempo de atendimento e tempo de viagem são dados de entrada previamente conhecidos. Em relação ao período de planejamento, aborda-se apenas o caso estático, onde um único período é considerado em todo horizonte de planejamento. Para

---

<sup>1</sup> termo usado para referir-se a algoritmos híbridos que integram meta-heurísticas e programação matemática

considerar a variabilidade no tempo de viagem que ocorre ao longo do horizonte de planejamento, este é dividido em intervalos de mesma duração, dentro dos quais a velocidade de deslocamento dos veículos é considerada constante. Além disso, os problemas abordados envolvem uma única função objetivo e são discretos. A rede de potenciais instalações e de clientes constitui um conjunto finito de vértices.

Entre os modelos propostos há a integração de decisões táticas, estratégicas e operacionais. Na prática, essa integração pode limitar o contexto de aplicação dos modelos. Estes se aplicam quando há facilidade de trocar a localização de instalações e a frota de veículos. Ou ainda, quando assume-se que a demanda dos clientes e o tempo de atendimento é invariável ao longo do tempo; existe uma variabilidade considerável entre os tempos de deslocamento em um mesmo arco, ou entre diferentes arcos, ao longo do horizonte de planejamento e, além disso, quando o padrão de tráfego permanece relativamente estável ao longo do tempo.

Devido às características dos problemas tratados, não se verificou na literatura problemas idênticos que permitissem comparar o desempenho dos modelos matemáticos desenvolvidos ou das abordagens heurísticas propostas nesta pesquisa com outras técnicas de resolução.

## 1.5 ESTRUTURA DO TRABALHO

Esta tese está estruturada conforme segue. No Capítulo 2 apresenta-se uma revisão acerca do problema de localização e roteamento, onde explicam-se os conceitos e características relevantes da versão clássica, reportam-se as principais variantes e extensões, expõem-se as abordagens de resolução mais recentes da literatura e aborda-se ainda a dependência de tempo em problemas de roteamento. No Capítulo 3 define-se o problema de localização e roteamento dependente do tempo, propõem-se uma formulação matemática e um algoritmo metaheurístico de resolução, descrevem-se os testes computacionais realizados, apresentam-se os resultados, discute-se o desempenho das abordagens propostas e finaliza-se com uma conclusão. Da mesma forma, no Capítulo 4, estabelece-se o problema de localização e roteamento dependente do tempo com dimensionamento de frota heterogênea (PLRDFH-DT), propõem-se um modelo matemático e um algoritmo de resolução meta-heurístico, reportam-se os experimentos conduzidos e os resultados obtidos, analisa-se o desempenho dos métodos e encerra-se com a conclusão. Por fim, no Capítulo 5, apresentam-se as considerações finais e as perspectivas para trabalhos futuros.

## 2 REVISÃO DE LITERATURA

Neste capítulo, expõe-se a teoria acerca da versão clássica do problema de localização e roteamento e destacam-se as suas principais características. Em seguida, reportam-se as variantes e extensões do problema que receberam maior atenção na literatura, em especial, a extensão que envolve frota não limitada e mista. Também apresentam-se as pesquisas mais atuais da área e as abordagens de resolução utilizadas. Por fim, aborda-se a questão da dependência de tempo na tomada de decisão em problemas da literatura que envolvam roteirização.

### 2.1 O PROBLEMA DE LOCALIZAÇÃO E ROTEAMENTO

O problema de localização e roteamento explora a interdependência entre dois tipos de decisão: a localização de instalações e o roteamento de veículos (YU et al., 2010; PRODHON; PRINS, 2014; DUKKANCI; KARA; BEKTAŞ, 2019). Apesar de serem duas decisões de natureza diferente, uma estratégica e a outra operacional, as pesquisas desenvolvidas nas últimas décadas comprovam os benefícios da sua integração para a redução de custos (SALHI; RAND, 1989; MIN; JAYARAMAN; SRIVASTAVA, 1998; NAGY; SALHI, 2007).

A relação de dependência entre a localização de instalações e o custo de transporte já foi evidenciada há mais de 50 anos (VON BOVENTER, 1961; MARANZANA, 1964). O primeiro trabalho a considerar claramente essas duas decisões de forma integrada foi, possivelmente, o de Watson Gandy e Dohrn (1973). Ainda assim, os benefícios potenciais dessa integração só foram quantificados anos mais tarde. Salhi e Rand (1989) mostraram que a estratégia de tomar essas duas decisões de forma independente pode produzir uma solução subótima. E isso é válido mesmo quando a localização é uma decisão de longo prazo (SALHI; NAGY, 1999). Tais observações serviram de estímulo para impulsionar o crescimento das pesquisas em torno do problema de localização e roteamento e suas variações.

Em sua versão clássica, o PLR é definido a partir de três conjuntos: as potenciais instalações (ou depósitos), os clientes e a frota de veículos. Assume-se que os veículos estão disponíveis para todos os depósitos, ou seja, nenhuma instalação possui uma frota específica. Além disso, o problema precisa ainda atender as seguintes restrições (PRODHON; PRINS, 2014):

- cada rota deve iniciar e finalizar no mesmo depósito;
- cada veículo deve executar, no máximo, uma rota;
- cada cliente deve ser servido uma única vez e por um único veículo;
- a demanda total dos clientes atendidos por um veículo deve respeitar a capacidade deste;

- a demanda total dos clientes atribuídos a um depósito não deve exceder a capacidade do mesmo.

Complementando as principais características do problema clássico, acrescentam-se os seguintes aspectos (DREXL; SCHNEIDER, 2017): função objetivo única e escalar; conjuntos finitos de potenciais depósitos e de clientes; dados de entrada determinísticos; período de planejamento único; impossibilidade de transferência de carga para localizações intermediárias e; inexistência de considerações de estoques, tanto para depósitos quanto para clientes.

O objetivo do PLR é minimizar o custo total de toda a logística envolvida, fazendo com que pelo menos duas decisões sejam tomadas de forma integrada. São elas: i) determinar, dentro do conjunto potencial, quais depósitos serão abertos e; ii) definir quais rotas serão executadas pelos veículos para atender a demanda dos clientes, partindo desses depósitos.

Em um contexto de redes de distribuição urbanas, a decisão de localização normalmente está associada com a escolha de um candidato concreto a depósito, enquanto em aplicações que consideram longas distâncias, essa decisão, em geral, se refere às regiões promissoras ou áreas industriais. Além disso, em zonas urbanas, o planejamento das rotas costuma envolver um grande número de clientes, como reflexo da densidade populacional dessas áreas (SCHNEIDER; LÖFFLER, 2019).

Para que a interdependência entre as duas decisões ocorra, a seleção de depósitos não deve ser implicitamente determinada pelas decisões de roteamento. E isso procede quando, pelo menos, uma das situações a seguir ocorre (DREXL; SCHNEIDER, 2015):

- há custos associados com a utilização dos depósitos, ou;
- há um número exato, ou máximo, de depósitos a serem abertos, ou;
- a capacidade dos depósitos é limitada.

Além das características principais, em algumas aplicações do problema há subcaracterísticas que podem diferir, mas que não alteram a sua natureza (DREXL; SCHNEIDER, 2015). São elas:

- rede direcionada ou não direcionada;
- depósitos capacitados ou não capacitados;
- inclusão ou não de custos fixos de abertura dos depósitos;
- limitação ou não do tamanho da frota;
- limitação ou não da capacidade da frota;
- frota homogênea ou heterogênea.

Para complementar, do ponto de vista matemático, o PLR pode ser modelado como um problema de otimização combinatória. Ele é classificado como NP-difícil, uma vez que possui uma clara ligação com dois problemas clássicos da literatura, amplamente conhecidos por essa classificação: o problema de localização de facilidades (PLF) e o problema de roteamento de veículos (PRV) (NAGY; SALHI, 2007).

### 2.1.1 Variações e extensões do problema

A necessidade de considerar diferentes aspectos na formulação dos modelos e adaptá-los às necessidades específicas impulsionou o crescimento da pesquisa, fazendo com que diferentes variações e extensões do PLR fossem propostas. As principais características que, segundo Drexler e Schneider (2015), alteram a natureza do problema e podem definir novas variantes são descritas na sequência.

Em relação às informações do problema, ele pode ser determinístico, estocástico ou difuso (*fuzzy*). A maior parte dos trabalhos da literatura é determinístico, ou seja, os dados são conhecidos com antecedência. O problema é classificado como estocástico quando alguma informação é dada a partir de uma distribuição de probabilidade, tal como em Marinakis, Marinaki e Migdalas (2016); e difuso, caso algum dado do problema esteja disponível de forma imprecisa, conforme reportado em Nadizadeh e Kafash (2019), por exemplo.

Quanto aos períodos de planejamento, pode-se ter um problema estático, dinâmico ou periódico. Quando o período de planejamento é único, tem-se o caso estático, que ocorre com maior frequência na literatura. No caso dinâmico, há múltiplos períodos e alguma informação inicialmente desconhecida torna-se disponível ao longo do tempo, como abordado em Gao et al. (2016). Por fim, se houver múltiplos períodos, mas todas as informações são previamente conhecidas, tem-se um problema periódico, tal como reportado em Koç (2016).

No que tange à rede de potenciais instalações, o problema pode ser classificado como discreto ou contínuo. Na primeira situação, e mais habitual, os potenciais depósitos representam um conjunto finito de vértices. Na segunda, por sua vez, o depósito pode ser instalado livremente em algum lugar do plano. Esta situação é considerada no trabalho de Manzour al Ajdad, Torabi e Salhi (2012).

Em relação aos níveis de abrangência das redes de distribuição, o problema pode ter um único nível, como é o caso do PLR clássico, onde há ligação apenas entre depósitos e clientes, ou múltiplos níveis. Neste último caso, os clientes não são atendidos diretamente de um depósito central, mas através de um rede de distribuição com  $N$  diferentes níveis, tal como estudado em Dai et al. (2019).

O problema pode ainda receber uma classificação quanto à função objetivo. Na maioria dos casos, ela tem um único objetivo, mas é possível que tenha múltiplos, assim como aborda Wang, Yang e Lu (2018), por exemplo. Além disso, no que diz respeito à estrutura das rotas, em geral, o roteamento dos veículos acontece em função dos vértices, que representam depósitos

e clientes. No entanto, é possível também que ele ocorra em função dos arcos, conforme considerado no trabalho de Doulabi e Seifi (2013).

Dentro desse contexto, há ainda na literatura algumas variações do problema que se destacam pela ampla aplicabilidade, e têm recebido atenção especial. Esse é o caso, por exemplo, do problema de localização e roteamento com coleta e entrega (*Location Routing Problem with Pickup and Delivery*). Nele, as tarefas a serem executadas podem englobar a entrega de mercadorias aos clientes, vindas de uma das instalações, ou a retirada de mercadorias nos clientes para entrega em uma das instalações, ou ainda ambas. Abordagens recentes desse problema foram feitas por Nadizadeh e Kafash (2019) e Capelle et al. (2019).

O problema de localização e roteamento com controle de estoque (*Inventory Routing Problem*), tratado em Alvarez, Munari e Morabito (2018) e Bertazzi et al. (2019), também é uma variante bastante estudada na literatura. Ela integra decisões de gerenciamento de estoque nas instalações, ou seja, avalia a quantidade de mercadoria que deve ser mantida em estoque, bem como a quantidade que deve ser solicitada ao fornecedor e em que momento isso deve ocorrer.

Além das variantes já mencionadas, pode-se destacar ainda o problema de localização e roteamento com 2 níveis (*Two-Echelon Location Routing Problem*). Em uma rede de distribuição com 2 níveis, 3 categorias de facilidades são consideradas. O transporte ocorre entre o primeiro e segundo níveis e, também, entre o segundo e terceiro níveis. Para cada um deles há veículos dedicados, que podem atender apenas as facilidades nele definidas. Além disso, transferências de carga são exigidas entre veículos de diferentes níveis. Esse tipo de problema é estudado por Breunig et al. (2016) e Darvish et al. (2019), entre outros autores.

Uma grande variedade de trabalhos aborda ainda diferentes variações simultaneamente, além de aplicações bem específicas. Como exemplo, pode-se citar a aplicação do PLR na logística de atendimentos de saúde, proposta em Veenstra et al. (2018). Nela, os autores abordam a entrega de medicamentos de uma farmácia, através da localização de guarda-volumes. Pacientes localizados dentro da distância de cobertura de um guarda-volume podem coletar seus medicamentos, enquanto os demais são atendidos por um serviço de entrega, feito em domicílio. O objetivo é determinar em quais dos potenciais locais os guarda-volumes devem ser alocados e quais rotas devem ser executadas para visitar os clientes que estão fora da área de cobertura, minimizando os custos de roteamento e de instalação dos guarda-volumes.

Outro exemplo é proposto por Luan et al. (2019), que aplica o PLR para auxiliar na tomada de decisão dentro da logística de emergência. O modelo busca relacionar a localização de instalações e as rotas de entrega, com o objetivo de maximizar a eficiência de resgate. Em geral, problemas desse tipo têm características de otimização de tempo, variabilidade dinâmica, complexidade ambiental e diversidade de suprimentos.

Em relação às extensões do problema de localização e roteamento, a literatura também reporta um número considerável de pesquisas. As principais extensões envolvem aspectos relacionados à diversificação da frota, especificidades no atendimento aos clientes, tais como

janelas de tempo, ou ainda custos associados com consumo de combustível, emissão de gases, utilização de veículos elétricos, entre outros.

Toro et al. (2017), por exemplo, propõem um PLR que considera emissões de gases de efeito estufa, onde o objetivo é minimizar o consumo de combustível. Essa extensão é denominada PLR Verde ou *Green Location Routing Problem*. Schiffer e Walther (2017) apresentam uma abordagem para o PLR que considera o roteamento de veículos elétricos e decisões de localização para estações de carregamento simultaneamente. Os autores consideram funções objetivo alternativas, de forma a minimizar não só a distância percorrida, como também o número de veículos necessários e de estações de carregamento, além dos custos totais. Problemas como esse são conhecidos na literatura como *Electric Location Routing Problem*.

O trabalho de Koç et al. (2016) aborda uma extensão do PLR que envolve janela de tempo para atendimento aos clientes e frota mista e não limitada. Nele, cada cliente deve ser atendido por um único veículo, definido entre os tipos disponíveis, dentro de um intervalo pré-estabelecido do horizonte de planejamento. Esse problema é conhecido como PLR com janela de tempo e dimensionamento de frota heterogênea ou *Fleet Size and Mix Location Routing Problem with Time Windows*.

O trabalho de revisão proposto por Drexler e Schneider (2015) contempla informações mais detalhadas sobre variantes e extensões do PLR.

#### 2.1.1.1 Problema de localização e roteamento com dimensionamento de frota heterogênea

Este problema é uma generalização do problema de roteamento de veículos com dimensionamento de frota heterogênea, ou *Fleet Size and Mix Vehicle Routing Problem* (GOLDEN et al., 1984), e uma extensão do PLR. Ele considera a decisão tática de compor uma frota apropriada de veículos para executar as tarefas de transporte, identificando assim o tamanho e os tipos de veículos que serão usados na execução das rotas. Para isso, assume-se uma frota não limitada, com diferentes capacidades, onde podem incidir custos fixos relacionados com a utilização de cada veículo. Considera-se também um conjunto de potenciais depósitos, nos quais podem recair custos de abertura e limitação de capacidade, e um conjunto de clientes com demandas que devem ser atendidas.

O problema consiste em abrir um subconjunto de depósitos, designar clientes a eles e determinar um conjunto de rotas considerando os diferentes tipos de veículos disponíveis, de forma que cada cliente seja servido por exatamente um deles. O veículo deve iniciar e terminar sua rota no respectivo depósito e as demandas dos clientes por ele atendidos devem respeitar a sua capacidade. O objetivo é encontrar a melhor composição da frota para atender os clientes, minimizando o custo total, que, em geral, é constituído por três componentes: custo fixo de abertura dos depósitos, custo fixo de utilização dos veículos e custo de execução das rotas. Ao final, definem-se os depósitos que serão utilizados, as rotas que serão executadas a partir deles e o tipo de veículo usado em cada rota.

Na sequência, abordam-se alguns trabalhos da literatura atual que englobam questões de localização e roteamento, e contemplam frota heterogênea.

O trabalho de Ambrosino, Sciomachen e Scutellà (2009) indiretamente considera frota heterogênea em um contexto de PLR. O estudo foi motivado por um aplicativo de uma empresa que opera no norte da Itália. Os autores propõem uma heurística de duas fases, que primeiro determina uma solução inicial factível e depois a aprimora usando técnicas de larga busca em vizinhança.

Koç et al. (2015) também estudam o PRV com frota heterogênea e com inclusão de janelas de tempo. Os autores desenvolveram um algoritmo evolutivo híbrido para resolver o problema, considerando tempo ou distância e usando tanto frota limitada quanto não limitada. Em outro trabalho, os mesmos autores também desenvolveram uma formulação baseada em programação linear inteira mista e um conjunto de desigualdades válidas para tratar o PLR com dimensionamento de frota heterogênea e janelas de tempo (KOÇ et al., 2016). Propuseram ainda um algoritmo de busca evolucionário híbrido para resolver o problema, que combina diversas meta-heurísticas.

Uma variante do problema multiobjetivo de localização e roteamento verde, onde os veículos podem iniciar e finalizar as rotas em depósitos distintos, é apresentada por Rabbani, Davoudkhani e Farrokhi-Asl (2017). O objetivo é definir as rotas que devem ser executadas pelos veículos para atendimento dos clientes, minimizando o custo total, que é composto pelo custo fixo do veículo e pelo custo de viagem baseado em distância e emissões de gás carbônico. Considerando questões ambientais, os autores propõem uma formulação matemática e um algoritmo evolutivo multiobjetivo para resolver o problema.

Um problema de localização e roteamento em dois níveis capacitado, que considera frota heterogênea e práticas de entrega conjunta, é abordado por Zhao, Eang e Souza (2018). Nele, o objetivo é determinar o conjunto de depósitos intermediários que deve ser aberto e a alocação de terminais de logística, com o intuito de minimizar o custo total de instalação e de execução conjunta de rotas. Os autores propõem um algoritmo heurístico de aproximação cooperativa, que é avaliado considerando entregas conjuntas e separadas, emissões de carbono e variações de demanda.

Em relação ao dimensionamento de frota heterogênea em um contexto de roteamento de veículos, pode-se mencionar também o trabalho de Lahyani, Coelho e Renaud (2018). Ele trata do problema de roteamento de veículos multidepósitos com dimensionamento de frota heterogênea. Os autores propõem cinco formulações distintas e desigualdades válidas, além de um algoritmo baseado em *branch-and-cut* e *branch-and-bound*.

Wang et al. (2019) introduzem um problema multiobjetivo de localização e roteamento com dimensionamento de frota heterogênea e janela de tempo. Duas versões de um algoritmo de múltipla busca em vizinhança multiobjetivo são propostas, as quais baseiam-se em decomposição e ângulo vetorial. Três diferentes operadores de busca integram esse algoritmo.



### 2.1.2 Abordagens de resolução da literatura

O problema de localização e roteamento tem sido estudado há décadas na literatura e as pesquisas em torno dele têm crescido bastante nos últimos anos. A grande maioria envolve abordagens de resolução heurísticas, enquanto métodos exatos e limites inferiores têm recebido menor atenção. Prodhon e Prins (2014) sugerem que isso se deve, possivelmente, ao fato do PLR ser um problema do tipo NP-difícil.

Limites inferiores para o PLR foram propostos por Barreto (2004), Albareda Sambola, Diaz e Fernandez (2005) e Baldacci, Mingozzi e Wolfler Calvo (2011). Entre as primeiras abordagens exatas desenvolvidas está o trabalho de Laporte e Nobert (1981), que propõem um algoritmo de resolução do tipo *branch-and-bound* para um caso específico que envolve a abertura de um único depósito. Em relação às pesquisas mais recentes destacam-se os trabalhos de Belenguer et al. (2011), que propõem um algoritmo de resolução exato do tipo *branch-and-cut*, baseado em um modelo de programação linear binária; e também de Baldacci, Mingozzi e Wolfler Calvo (2011) e Contardo, Cordeau e Gendron (2014a), em que uma abordagem exata baseada em corte e geração de colunas é proposta. Ambos baseiam seus métodos na decomposição do problema em um conjunto de problemas de roteamento de veículos com múltiplos depósitos.

No que tange aos métodos heurísticos, uma grande variedade de abordagens tem sido proposta. Entre elas, mencionam-se as principais a seguir.

*Adaptive Large Neighborhood Search* (ALNS) é uma abordagem caracterizada pela destruição e reparo da solução corrente, a partir da utilização de heurísticas de remoção e inserção. Em cada iteração, a escolha das heurísticas é realizada através de um peso ajustado dinamicamente, com base na eficiência de cada método ao longo da busca. Ela foi aplicada, por exemplo, no trabalho de Hemmelmayr, Cordeau e Crainic (2012).

Fundamentada por uma analogia com a termodinâmica, *Simulated Annealing* (SA) é uma meta-heurística para otimização que consiste em uma técnica de busca local probabilística. No contexto do PLR, ela foi usada por Yu et al. (2010), entre outros autores. Já a *Ant Colony Optimization* (ACO) é uma heurística inspirada no comportamento de formigas. Ela é baseada em probabilidade, e foi criada para solucionar problemas computacionais que envolvem a busca de caminhos em grafos, tal como aplicado no trabalho de Ting e Chen (2013).

*Variable Neighborhood Search* (VNS) é um método meta-heurístico que explora vizinhanças distantes da solução corrente, movendo-se de um local para outro apenas quando uma melhoria é gerada. Esse método de busca local é aplicado repetidamente para obtenção de ótimos locais, a partir de buscas na vizinhança. Ele foi utilizado na pesquisa de Escobar et al. (2014), por exemplo. Uma rotina de pesquisa local que também pode ser citada, normalmente usada em outras meta-heurísticas, é a chamada *Variable Neighborhood Descent* (VND). Ela explora várias estruturas de vizinhança de maneira determinística e seu sucesso é baseado no fato que diferentes estruturas geralmente não têm o mesmo mínimo local. Foi aplicada, por exemplo, no trabalho de Pekel e Kara (2018).

A abordagem *Granular Tabu Search* (GTS) utiliza uma ferramenta de intensificação e diversificação, baseada no uso de bairros drasticamente restritos, também chamados granulares. Ela não realiza movimentações que envolvem apenas elementos que, provavelmente, não pertencem a boas soluções factíveis. Os bairros restritos podem ser vistos como uma implementação eficiente de estratégias de lista de candidatos propostas para algoritmos de busca tabu. Um exemplo de tal aplicação está no trabalho desenvolvido por Schneider e Löffler (2019);

*Greedy Randomized Adaptive Search Procedure* (GRASP) é um método construtivo que consiste em criar uma solução inicial e depois efetuar uma busca local para melhorar a qualidade da solução. Seu diferencial em relação aos outros métodos está na geração dessa solução, que é baseada em uma busca gulosa, aleatória e adaptativa, conforme o nome sugere. Essa meta-heurística foi empregada, no contexto do PLR, por Duhamel et al. (2010) e Contardo, Cordeau e Gendron (2014b), entre outros.

O algoritmo *Evolutionary Local Search* (ELS) é um método que combina busca local com algoritmos evolutivos. A busca local se concentra em encontrar soluções promissoras, enquanto o algoritmo evolutivo realiza a tarefa de gerar novas soluções através de seleção, cruzamento e mutação. Tal técnica foi aplicada por Duhamel et al. (2010).

Além disso, há ainda abordagens matemáticas que integram técnicas de *Instruction-level parallelism* (ILP), ou seja, técnicas para que instruções de um programa de computador possam ser executadas simultaneamente, tais como aplicado em Prins et al. (2007), Escobar, Linfati e Toth (2013) e Contardo, Cordeau e Gendron (2014b).

Revisões de literatura mais detalhadas acerca do problema de localização e roteamento foram elaboradas Lopes et al. (2013), Prodhon e Prins (2014) e Drexler e Schneider (2017).

## 2.2 A DEPENDÊNCIA DO TEMPO EM PROBLEMAS DE ROTEAMENTO

Conforme já discutido anteriormente, em redes de distribuição urbanas, a premissa de tempo de deslocamento constante não é, de maneira geral, verdadeira. No entanto, na maioria dos estudos sobre integração de localização e roteamento, a questão operacional é tratada em termos de distância e não de tempo. Uma notável exceção é o trabalho de Hu et al. (2018), que considera tráfego e número de caminhões na rota dependentes um do outro. Os autores trabalham com informações reais sobre tráfego e as decisões táticas e operacionais são direcionadas a evitar o congestionamento do tráfego existente.

No contexto do problema de roteamento de veículos, o interesse em incluir a dependência de tempo nas questões de roteamento tem aumentado a partir da publicação do trabalho de Malandraki e Daskin (1992). Nele, os autores identificam duas fontes de flutuação que afetam o tempo de viagem: uma se deve à variação temporal do tráfego e a outra é aleatória, causada por condições climáticas, acidentes, entre outros fatores. Eles ainda argumentam que a variação temporal pode ser representada por um modelo determinístico, onde o tempo de viagem é definido não só pela distância, mas também pela hora do dia.

Posteriormente, a literatura científica reconheceu que tais modelos são de pouca importância se não houver dados reais para apoiar as decisões (ICHOUA; GENDREAU; POTVIN, 2003). No trabalho de Ichoua, Gendreau e Potvin (2003), os autores propõem um modelo com velocidade dependente do tempo, que satisfaz a propriedade *first in first out* (FIFO). Ele inclui também janelas de tempo e tempo de serviço nos vértices, e funções degrau são utilizadas para calcular o tempo de deslocamento dos veículos. Os autores desenvolvem uma formulação usando programação linear inteira mista, além de uma heurística baseada em vizinho mais próximo e um algoritmo de ramificação e corte, os quais são usados para resolver problemas estáticos e dinâmicos.

Mais recentemente, em um estudo sobre o problema do caixeiro viajante dependente do tempo, Cordeau, Ghiani e Guerriero (2014) apontaram que quanto mais semelhança houver no padrão de flutuação dos tempos de deslocamento dos arcos, mais a solução do problema, obtida a partir de tempos de viagem constantes, também é precisa para o problema com dependência de tempo. Se todos os arcos da rede compartilham exatamente o mesmo padrão, a solução ótima é a mesma para os dois problemas.

Um problema de roteamento de veículos dependente do tempo, que considera níveis de congestionamento, é estudado por Mancini (2017). A autora propôs uma abordagem de solução de duas fases. Inicialmente, uma heurística construtiva aleatória gera múltiplas rotas e, em seguida, essas rotas são usadas como entrada em uma formulação de particionamento de conjuntos. Os resultados apontam que considerar a variabilidade nos tempos de viagem, causada pelo congestionamento, permite obter melhores planos de execução de rotas.

Alvarez et al. (2018) incorporam custos de congestionamento em tempo real ao problema de roteamento de veículos. Os autores comparam os resultados obtidos por quatro funções objetivo diferentes, que envolvem minimização de distância euclidiana, distância real, tempo real com congestionamento estático e tempo real com congestionamento dinâmico. Eles destacam os custos de congestionamento como um importante fator na otimização da distribuição.

Jaballah et al. (2019) formulam um problema que integra caminho mais curto e roteamento de veículos, com dependência de tempo. A formulação é testada em instâncias geradas a partir de dados reais, e os resultados também mostram a importância de considerar o congestionamento do tráfego na otimização de rotas de entrega.

De acordo com Akyol e Koster (2018), o uso de informações de tráfego para impulsionar novas maneiras de efetuar entregas não afeta apenas as decisões operacionais de roteamento. Afeta também a forma de oferecer possíveis tempos de entrega aos clientes e de acessar determinadas áreas da cidade. Além disso, o uso de caminhos alternativos, para evitar o congestionamento do tráfego, pode levar a reduções significativas nas emissões de gases de efeito estufa (BEHNKE; KIRSCHSTEIN, 2017). Usando uma abordagem semelhante, Marufuzzaman e Ekşioğlu (2017) mensuram seus ganhos em termos financeiros.

Informações complementares e mais abrangentes sobre modelagem de problemas de roteamento com dependência de tempos e métodos de solução podem ser encontradas no trabalho desenvolvido por Gendreau, Ghiani e Guerriero (2015).

### 2.2.1 Considerações sobre a propriedade FIFO

Na utilização de dados reais de tráfego, deve-se observar que as redes de distribuição respeitam inerentemente a propriedade FIFO. A mesma condição não se aplica necessariamente quando os tempos de deslocamento são discretizados em intervalos.

Em diversos trabalhos da literatura, os autores não explicitam se tal propriedade é respeitada (MILLER-HOOKS; YANG, 2005; EHMKE; CAMPBELL; THOMAS, 2016). Caso ela não seja considerada, os algoritmos exigem etapas adicionais para evitar a formação de ciclos.

Dentro desse contexto, Sung et al. (2000) introduziram o modelo de velocidade de fluxo, ou *Flow Speed Model* (FSM), que garante que redes de distribuição dependentes do tempo respeitem a propriedade FIFO. O modelo calcula a velocidade de fluxo para cada porção de um segmento, dependendo do momento em que este é atravessado. Assim, se o limite entre dois intervalos é ultrapassado, a velocidade é atualizada e os tempos de deslocamento são calculados de acordo com ela.

Um método alternativo é o modelo de ligação de tempo de viagem, ou *Link Travel Time Model* (LTM) (DELLING; NANNICINI, 2012). Nele, o tempo de deslocamento no arco é fixado pelo momento em que ele começa a ser atravessado. Assim, dentro de um intervalo pré-estabelecido, o tempo de deslocamento se mantém constante.

### 3 PROBLEMA DE LOCALIZAÇÃO E ROTEAMENTO DEPENDENTE DO TEMPO

Neste capítulo, introduz-se o problema de localização e roteamento dependente do tempo (SCHMIDT et al., 2019). Na Seção 3.1, define-se formalmente o problema e propõe-se uma formulação matemática a partir de um modelo de programação linear inteira mista. Na Seção 3.2, apresenta-se também uma abordagem de resolução heurística, cuja estrutura descreve-se em detalhes. Na Seção 3.3, especificam-se os procedimentos adotados na condução dos testes computacionais e na geração das instâncias. Na Seção 3.4, apresentam-se os resultados de extensivos testes computacionais e discutem-se os mesmos. Por fim, na Seção 3.5, apresentam-se as conclusões.

#### 3.1 DESCRIÇÃO DO PROBLEMA E FORMULAÇÃO MATEMÁTICA

O PLR-DT é definido em um grafo direcionado  $G(V, A)$ , onde  $V$  representa o conjunto de vértices e  $A$  o conjunto de arcos. Sejam  $V_d$  o conjunto de todos os potenciais depósitos,  $V_c$  o conjunto de todos os clientes e  $V_t$  um conjunto de vértices fictícios, denominados terminais, tal que  $V_d \cap V_c = \emptyset$ ,  $V_d \cap V_t = \emptyset$ ,  $V_c \cap V_t = \emptyset$  e  $V = V_d \cup V_c \cup V_t$ . Definem-se ainda  $K$  como um conjunto de veículos homogêneos, com capacidade individual limitada em  $Q$  unidades; e  $\delta(i)$  como um subconjunto de  $|K|$  terminais associado ao depósito  $i \in V_d$ , ou seja,  $\delta(i) \subseteq V_t$ . Cada terminal  $v \in \delta(i)$ ,  $i \in V_d$  está associado a um veículo da frota e indica o retorno deste ao depósito. A utilização de terminais possibilita o controle dos veículos da frota usados na execução das rotas, bem como da capacidade usada em cada rota e do tempo dispendido na sua execução.

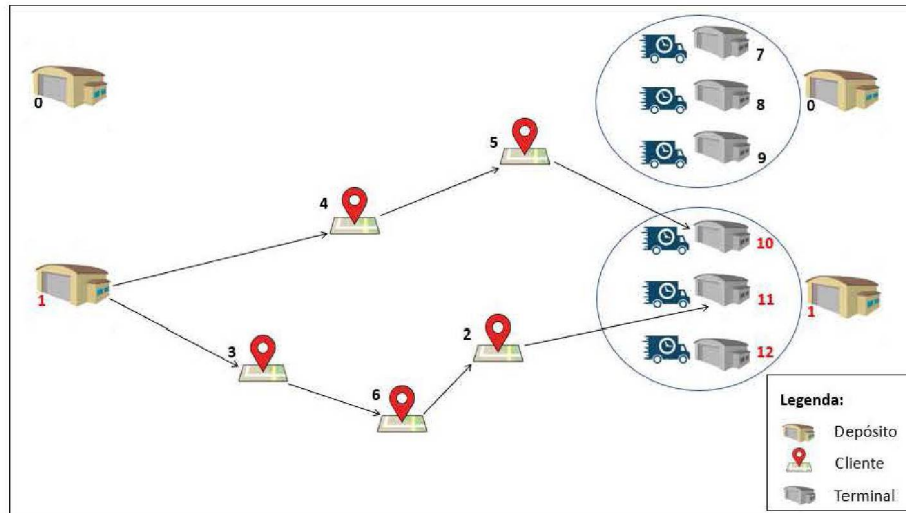
Sejam  $A_{dc}$ ,  $A_{cc}$ ,  $A_{ct}$  conjuntos de arcos, tais que cada arco  $(i, j)$  é dado a partir de um dos produtos cartesianos  $A_{dc} = V_d \times V_c$ ;  $A_{cc} = V_c \times V_c$ ,  $i \neq j$ , e  $A_{ct} = \cup_{i \in V_d} \{V_c \times \delta(i)\}$ , de forma que  $A = A_{dc} \cup A_{cc} \cup A_{ct}$ . Seja ainda  $H$  o conjunto de intervalos de tempo. Define-se intervalo como um período de tempo para o qual o padrão de tráfego é constante. A cada arco  $(i, j) \in A$  está associado um tempo de deslocamento  $t_{ij}^h$ , que depende do intervalo  $h \in H$ . O horizonte de planejamento é de um dia, ou uma fração do dia, dividido em intervalos de tempo iguais, com duração de  $\bar{T}$  unidades. Para  $h \in H = \{0, 1, \dots, h, \dots, m\}$ ,  $m + 1$  representa o número de intervalos diários. Dessa forma,  $[h\bar{T}, (h + 1)\bar{T} - \varepsilon]$  representa o intervalo associado a  $h$ , onde  $\varepsilon$  é um número positivo que indica uma unidade de tempo.

A demanda de cada cliente  $i \in V_c$  é denotada por  $q_i$  e o tempo de serviço nele por  $s_i$ . Sabendo que todos os clientes precisam ser atendidos a partir de um único depósito e considerando uma frota homogênea, limitada e horário fixo de saída dos veículos no depósito, o objetivo é minimizar o tempo total de deslocamento. Ao final, identificam-se o depósito com a melhor localização e as rotas que devem ser executadas para atendimento dos clientes.

Para ilustrar a função dos terminais, considere como exemplo um problema com dois potenciais depósitos, cinco clientes e uma frota homogênea, limitada em três veículos. Nele há

um terminal associado com cada veículo da frota, para cada potencial depósito. Uma solução para esse problema é representada através da Figura 3.1.

FIGURA 3.1 - REPRESENTAÇÃO DA SOLUÇÃO DE UM PLR-DT



FONTE: A autora (2019)

Note que o terminal serve para indicar o retorno de um veículo ao depósito de origem. No exemplo em questão, dois veículos da frota são usados para executar as rotas e atender os clientes, de forma que, para cada um desses veículos, um terminal associado ao depósito de origem também foi acessado. Isso possibilita ainda controlar o tempo de retorno de cada veículo através do terminal, bem como a demanda a ele designada.

Para o modelo matemático, definem-se as seguintes variáveis:

- a variável binária  $x_{ij}^h$  assume valor 1 se o arco  $(i, j)$  é atravessado por um veículo no intervalo de tempo  $h$ , e 0 em caso contrário;
- a variável binária  $z_{ij}$  assume valor 1 se o arco  $(i, j)$  é atravessado por um veículo, e 0 em caso contrário;
- a variável binária  $w_i$  assume valor 1 se o depósito  $i$  é aberto, e 0 em caso contrário;
- a variável binária  $y_i^h$  assume valor 1 se uma rota parte do vértice  $i \in V \setminus V_t$  no intervalo de tempo  $h \in H$ , e 0 em caso contrário;
- a variável contínua  $a_i$  indica o momento de partida nos vértices correspondentes aos depósitos ou clientes,  $i \in V \setminus V_t$ , ou de chegada, caso o vértice seja um terminal  $i \in V_t$ ;
- a variável contínua  $u_i$  indica a demanda acumulada entregue até a saída do veículo no cliente  $i \in V_c$ , caso essa demanda seja igual a capacidade do veículo. No entanto, se for inferior, essa variável pode assumir uma folga. Nesse caso, ela pode admitir qualquer valor dentro do intervalo correspondente a demanda total entregue e a capacidade do veículo.

A Tabela 3.1 resume a notação usada na formulação matemática do problema.

Tabela 3.1: NOTAÇÃO USADA NO PLR-DT

Parâmetros	
$m + 1$	Número de intervalos de tempo do horizonte de planejamento
$\bar{T}$	Duração de cada intervalo de tempo $h \in H$
$\varepsilon$	Uma unidade de tempo considerada
$q_i$	Demanda de cada cliente $i \in V_c$
$s_i$	Tempo de serviço no cliente $i \in V_c$
$Q$	Capacidade de cada veículo da frota
$t_{ij}^h$	Tempo de viagem no arco $(i, j) \in A$ , no intervalo de tempo $h \in H$
Conjuntos	
$V_c$	Conjunto de clientes
$V_d$	Conjunto de potenciais depósitos
$V_t$	Conjunto de terminais
$K$	Conjunto de veículos da frota
$H$	Conjunto de intervalos de tempo
$A$	Conjunto de arcos
$\delta(i)$	Subconjunto de $ K $ terminais para cada potencial depósito $i \in V_d$
Variáveis de decisão	
$x_{ij}^h$	Se um veículo atravessa, ou não, o arco $(i, j) \in A$ no intervalo de tempo $h \in H$
$z_{ij}$	Se um veículo atravessa, ou não, o arco $(i, j) \in A$
$w_i$	Se o depósito $i \in V_d$ é aberto, ou não
$y_i^h$	Se algum veículo parte, ou não, do vértice $i \in V \setminus V_t$ no intervalo de tempo $h \in H$
$a_i$	Momento de partida no vértice, para $i \in V \setminus V_t$ , ou de chegada no vértice, para $i \in V_t$
$u_i$	Demanda acumulada entregue na saída do veículo no cliente $i \in V_c$ , podendo admitir uma folga
Índices	
$h$	Intervalo de tempo
$i, j, v$	Vértices

FONTE: A autora (2019)

A formulação matemática proposta é descrita na sequência:

$$\min \sum_{(i,j) \in A} \sum_{h \in H} t_{ij}^h x_{ij}^h \quad (3.1)$$

sujeito a:

$$\sum_{i \in (V_c \setminus \{j\}) \cup V_d} z_{ij} = 1, \quad \forall j \in V_c \quad (3.2)$$

$$\sum_{j \in (V_c \setminus \{i\}) \cup V_t} z_{ij} = 1, \quad \forall i \in V_c \quad (3.3)$$

$$\sum_{j \in V_c} z_{ij} \leq |K| w_i, \quad \forall i \in V_d \quad (3.4)$$

$$\sum_{j \in V_c} z_{jv} \leq w_i, \quad \forall v \in \delta(i), \forall i \in V_d \quad (3.5)$$

$$(3.6) \quad \sum_{i \in V_d} w_i = 1$$

$$(3.7) \quad u_i - n_j + \tilde{Q} z_{ij} \leq \tilde{Q} - q_j, \quad \forall i, j \in V^c, i \neq j$$

$$(3.8) \quad q_i \leq u_i \leq \tilde{Q}, \quad \forall i \in V^c$$

$$(3.9) \quad \sum_{j \in V^c} x_{ij}^0 = \sum_{j \in V^c} \sum_{v \in \delta(i)} z_{fv}, \quad \forall i \in V_d$$

$$(3.10) \quad \sum_{h \in H} x_{ij}^h = z_{ij}, \quad \forall (i, j) \in A$$

$$(3.11) \quad x_{ij}^h \leq y_h^i, \quad \forall (i, j) \in A, \forall h \in H$$

$$(3.12) \quad \sum_{h \in H} y_h^i = w_i, \quad \forall i \in V_d$$

$$(3.13) \quad \sum_{h \in H} y_h^i = 1, \quad \forall i \in V^c$$

$$(3.14) \quad a_i = 0, \quad \forall i \in V_d$$

$$(3.15) \quad a_i \geq a_i + s_j + t_{ij}^h - 2\overline{T}(m+1)(1 - x_{ij}^h), \quad \forall (i, j) \in A, \forall h \in H$$

$$(3.16) \quad a_i \leq a_i + s_j + t_{ij}^h + \overline{T}(m+1)(1 - x_{ij}^h), \quad \forall (i, j) \in A, \forall h \in H$$

$$(3.17) \quad \sum_{h \in H} h \overline{T} y_h^i \leq a_i \leq \sum_{h \in H} h \overline{T} y_h^i + \overline{T} - \varepsilon, \quad i \in V^c$$

$$(3.18) \quad a_v \leq \sum_{j \in V^c} z_{fv} [\overline{T}(m+1) - \varepsilon], \quad \forall v \in \delta(i), \forall i \in V_d$$

$$(3.19) \quad w_i \in \{0, 1\}, \quad \forall i \in V_d$$

$$(3.20) \quad z_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A$$

$$(3.21) \quad y_h^i \in \{0, 1\}, \quad \forall i \in V \setminus V_d, \forall h \in H$$



$$x_{ij}^h \in \{0, 1\}, \quad \forall (i, j) \in A, \forall h \in H \quad (3.22)$$

$$a_i \in \mathbb{R}_+, \quad \forall i \in V \quad (3.23)$$

$$u_i \in \mathbb{R}_+, \quad \forall i \in V_c. \quad (3.24)$$

O problema consiste em minimizar o tempo total de deslocamento para execução das rotas, expresso por (3.1). As restrições (3.2) e (3.3) são restrições de designação, que garantem que cada cliente seja visitado exatamente uma vez. O conjunto de restrições (3.4) impõe que o número de rotas que parte de um depósito aberto seja, no máximo, igual ao tamanho da frota. Caso um depósito seja utilizado, as restrições (3.5) asseguram que cada terminal vinculado a ele pode ser utilizado, no máximo, uma vez. Note que há um terminal associado a cada veículo da frota, para cada potencial depósito. Assim, cada rota criada é finalizada em um dos terminais. A restrição (3.6) impõe a abertura de um único depósito. Restrições dos conjuntos (3.7) e (3.8) são uma extensão da eliminação de sub-rotas de Miller-Tucker-Zemlin, originalmente propostas por Kulkarni e Bhave (1985). As restrições (3.9) garantem que os veículos deixem o depósito no primeiro intervalo de tempo, além de equilibrar o fluxo que sai do depósito e que chega nos terminais associados. Um único intervalo de tempo é considerado na travessia de cada arco  $(i, j)$ , conforme impõem as restrições (3.10). Se um arco  $(i, j)$  é atravessado por um veículo no intervalo  $h$ , então, de acordo com as restrições (3.11),  $h$  é o intervalo de tempo considerado na saída do veículo do vértice  $i$ . Através do conjunto de restrições (3.12), a saída no depósito ocorre apenas se este estiver aberto. As restrições (3.13) garantem que apenas um intervalo de tempo  $h$  seja usado na saída do veículo do cliente  $i$ . O conjunto (3.14) fixa em zero o momento de saída dos veículos no depósito. Usando as restrições (3.15) e (3.16), é possível controlar o momento de saída dos veículos em cada vértice. Note que nas restrições do conjunto (3.15) é necessário multiplicar o último termo por 2 para garantir que ele ainda atue como um *Big-M* válido, caso  $a_i + s_j + t_{ij}^h$  seja maior que o horizonte de planejamento. O momento de saída no vértice  $j$ , após a travessia do arco  $(i, j)$ , inclui o momento de saída no vértice  $i$ , o tempo de deslocamento necessário para atravessar o arco  $(i, j)$  e o tempo de serviço no cliente  $j$ . As restrições (3.17) limitam o momento de saída no vértice  $i$  ao intervalo de tempo correspondente. Restrições do conjunto (3.18) asseguram que os veículos executem suas rotas dentro do horizonte de planejamento. Finalmente, as restrições (3.19)–(3.24) forçam as condições de integralidade e de não-negatividade das variáveis.

Note que as restrições (3.10) e (3.21) implicam  $x_{ij}^h \leq 1$ . Mais ainda, conforme veremos no teorema abaixo, a não-negatividade das variáveis  $x_{ij}^h$  já garante a integralidade, ou seja, assumindo não-negatividade das variáveis  $x_{ij}^h$ , as restrições (3.22) são redundantes.

**Teorema 3.1** *Suponha que as restrições (3.10) – (3.13) e (3.19) – (3.21) sejam satisfeitas. Se*

$$x_{ij}^h \geq 0, \quad \forall (i, j) \in A, \forall h \in H, \quad (3.25)$$

*então  $x_{ij}^h \in \{0, 1\}, \forall (i, j) \in A, \forall h \in H$ . Em particular, para todo depósito  $i \in V_d$  que não é aberto, tem-se que  $x_{ij}^h = 0, \forall j \in V_c, \forall h \in H$ .*

*Demonstração: Considere  $(i, j) \in A$  e  $h^* \in H$  arbitrários. Se  $y_i^{h^*} = 0$ , então  $x_{ij}^{h^*} = 0$  por (3.11) e (3.25). Caso contrário, por (3.21), tem-se que  $y_i^{h^*} = 1$ . Assim, de (3.12), (3.13), (3.19) e (3.21), segue que  $y_i^h = 0$ , e portanto  $x_{ij}^h = 0, \forall h \neq h^*$ . Logo, usando (3.10) e (3.20), obtém-se  $x_{ij}^{h^*} = \sum_{h \in H} x_{ij}^h = z_{ij} \in \{0, 1\}$ , provando a primeira afirmação do teorema. Para a outra afirmação, note que se  $i \in V_d$  não é aberto, então  $w_i = 0$ . Assim, por (3.12) e (3.21), segue que  $y_i^h = 0, \forall h \in H$ . Logo, usando (3.11) e (3.25), conclui-se que  $x_{ij}^h = 0, \forall j \in V_c, \forall h \in H$ .*

*Portanto,  $x_{ij}^h \in \{0, 1\}, \forall (i, j) \in A, \forall h \in H$ .*

### 3.1.1 Desigualdades válidas

Para fortalecer o modelo, define-se um conjunto de desigualdades válidas, a seguir descritas.

As restrições (3.7) podem ser ajustadas, como em Kara, Laporte e Bektaş (2004), produzindo (3.26):

$$u_i - u_j + Qz_{ij} + (Q - q_i - q_j)z_{ji} \leq Q - q_j, \quad \forall i, j \in V_c, i \neq j. \quad (3.26)$$

Conforme imposto pelas restrições (3.14), todas as saídas do depósito acontecem em  $h = 0$ . Dessa forma, é possível reduzir o tamanho do problema, eliminando algumas variáveis:

$$y_i^h = 0, \quad \forall i \in V_d, \forall h \in H \setminus \{0\} \quad (3.27)$$

$$x_{ij}^h = 0, \quad \forall i \in V_d, \forall j \in V_c, \forall h \in H \setminus \{0\}. \quad (3.28)$$

É possível que algumas variáveis, associadas com a saída dos veículos nos clientes, possam ser removidas. Isso acontece quando o tempo de serviço no cliente  $i$  somado ao menor tempo para atravessar o arco entre algum depósito (ou outro cliente) e o cliente  $i$  é maior que a duração do intervalo. Também ocorre se o tempo de serviço em um cliente somado ao menor tempo necessário para atravessar um arco entre algum depósito e algum cliente é maior que a duração do intervalo:

$$x_{ij}^0 = 0 \quad \forall i \in V_c \mid \left( \min_{a \in (V_c \setminus \{i\}) \cup V_d} \{t_{ai}^0\} + s_i \right) \geq \bar{T}, \quad \forall j \in V_c \cup V_t, j \neq i \quad (3.29)$$

$$x_{ij}^0 = 0 \quad \forall i \in V_c \mid s_i \geq \bar{T} - \min_{\substack{a \in V_d \\ b \in V_c}} \{t_{ab}^0\}, \quad \forall j \in V_c \cup V_t, j \neq i. \quad (3.30)$$

Tal ponderação pode ser generalizada para qualquer intervalo de tempo  $h \in H$ :

$$x_{ij}^h = 0 \quad \forall i \in V_c \mid \left( \min_{a \in (V_c \setminus \{i\}) \cup V_d} \{t_{ai}^h\} + s_i \right) \geq (h+1)\bar{T}, \quad \forall j \in V_c \cup V_t, j \neq i, \forall h \in H \quad (3.31)$$

$$x_{ij}^h = 0 \quad \forall i \in V_c \mid s_i \geq (h+1)\bar{T} - \min_{\substack{a \in V_d \\ b \in V_c}} \{t_{ab}^h\}, \quad \forall j \in V_c \cup V_t, j \neq i, \forall h \in H. \quad (3.32)$$

Da mesma forma, para o último intervalo de tempo também pode ser possível remover algumas variáveis do problema, associadas com os terminais. Isso ocorre quando o tempo necessário para atravessar um arco, no último intervalo, e chegar ao terminal é maior que a duração do intervalo. Ou ainda, se o menor tempo necessário para chegar ao cliente, somado com seu tempo de serviço e com o menor tempo para ir desse cliente ao terminal, é maior que a duração do intervalo.

$$x_{ij}^m = 0, \quad \forall i \in V_c, \forall j \in V_t \mid t_{ij}^m \geq \bar{T} \quad (3.33)$$

$$x_{ij}^m = 0, \quad \forall j \in V_c \mid \left( \min_{a \in V_c \setminus \{j\}} \{t_{aj}^m\} + s_j + \min_{b \in V_t} \{t_{jb}^m\} \right) \geq \bar{T}, \quad \forall i \in V_c, i \neq j. \quad (3.34)$$

É possível ainda estabelecer um limite inferior a partir de uma soma de menores tempos. Seja  $k'$  o número mínimo de veículos necessários para que a demanda dos clientes seja atendida, obtido como solução de um problema de empacotamento, e onde  $Q$  é a capacidade do veículo. Seja  $s_{di} = t_{di}^0, \forall d \in V_d, \forall i \in V_c$ . Para todo  $d \in V_d$ , seja  $s_{di_1} \mid s_{di_1} \leq s_{di_2} \leq \dots \leq s_{di_{|V_c|}}$ . Similarmente, seja  $r_{id} = \min_{h \in H} \{t_{id}^h\}$  e  $r_{id_n} \mid r_{id_1} \leq r_{id_2} \leq \dots \leq r_{id_{|V_c|}}$ . Agora, seja  $f_d = \sum_{n=1}^{k'} (s_{di_n} + r_{id_n}), \forall d \in V_d$ . Finalmente, seja  $g_c = \min_{\substack{a \in V_c \setminus \{i\} \\ h \in H}} \{t_{ca}^h\}, \forall c \in V_c$  e  $g_{c_n} \mid g_{c_1} \leq g_{c_2} \leq \dots \leq g_{c_{|V_c|}}$ .

$$\sum_{(i,j) \in A} \sum_{h \in H} t_{ij}^h x_{ij}^h \geq \min_{d \in V_d} \{f_d\} + \sum_{n=1}^{|V_c| - k'} g_{c_n}. \quad (3.35)$$

Em relação ao roteamento do modelo, uma melhoria que pode ser feita é evitar que sub-rotas com dois ou três clientes sejam criadas:

$$z_{ij} + z_{ji} \leq 1, \quad \forall i, j \in V_c, i \neq j \quad (3.36)$$

$$z_{ij} + z_{ji} + z_{iv} + z_{vi} + z_{jv} + z_{vj} \leq 2, \quad \forall i, j, v \in V_c, i \neq j, i \neq v, j \neq v. \quad (3.37)$$

As desigualdades (3.38) estabelecem que o número de veículos saindo do depósito é igual ao número de veículos chegando aos terminais a ele associados.

$$\sum_{j \in V_c} z_{ij} = \sum_{v \in \delta(i)} \sum_{j \in V_c} z_{jv}, \quad \forall i \in V_d. \quad (3.38)$$

Os fluxos de saída do depósito são controlados pelas desigualdades (3.39) e (3.40). Para o depósito selecionado, o resultado de um problema de empacotamento determina o número mínimo de veículos ( $k'$ ), definido como limite inferior para todas as saídas em  $h = 0$ . Esse resultado também determina o total de veículos que parte de algum depósito.

$$\sum_{j \in V_c} x_{ij}^0 \geq w_i k', \quad \forall i \in V_d \quad (3.39)$$

$$\sum_{i \in V_d} \sum_{j \in V_c} z_{ij} \geq k'. \quad (3.40)$$

O intervalo de tempo, usado na solução, deve ser o mesmo para as variáveis  $y_i^h$  e  $x_{ij}^h$ :

$$\sum_{j \in V_c \setminus \{i\} \cup V_t} x_{ij}^h = y_i^h, \quad \forall i \in V_c, \forall h \in H. \quad (3.41)$$

Finalmente, é possível evitar algumas soluções simétricas impondo uma ordem de uso dos terminais. Para todo  $v \in \delta(i)$ ,  $i \in V_d$ , seja  $v_n | v_1 \leq v_2 \leq \dots \leq v_{|K|}$ .

$$\sum_{j \in V_c} z_{jv} \geq w_i, \quad \forall v \in \delta(i) \mid v_1 \leq v \leq v_{k'}, \quad \forall i \in V_d \quad (3.42)$$

$$a_v \geq a_{v+1}, \quad \forall v \in \delta(i) \mid v_1 \leq v \leq v_{k'} \wedge k' < |K|, \quad \forall i \in V_d. \quad (3.43)$$

## 3.2 ALGORITMO MATEURÍSTICO

A estrutura geral da abordagem mateurística proposta é composta de duas fases. A primeira engloba a geração de um conjunto formado por rotas individuais de veículos. Ele é gerado a partir de um grupo de heurísticas construtivas. Na segunda fase, essas rotas são processadas e então usadas como potenciais variáveis em um modelo de particionamento de conjuntos. Cada uma dessas fases é descrita em detalhes na sequência.

### 3.2.1 Heurísticas construtivas

Sete heurísticas construtivas são usadas para gerar rotas de veículos, a partir de cada potencial depósito. As rotas geradas são sempre factíveis, tanto no que diz respeito à capacidade de atender os clientes dentro do horizonte de planejamento, como em relação à capacidade do

veículo. O intuito principal é que, a cada execução de uma heurística, todos os clientes tenham sido designados a uma das rotas geradas. Se isso ocorre, uma solução factível e completa é criada e então adicionada a um conjunto de soluções completas, denominado  $\Lambda$ . Caso a heurística falhe em sua tarefa, deixando de designar algum cliente a alguma rota, a solução gerada é adicionada ao conjunto de soluções parciais  $\lambda$ .

Dado o aspecto aleatório envolvido nesses procedimentos construtivos, eles são repetidos diversas vezes até que um critério de parada seja atendido. Conforme mostra o Algoritmo 1, a execução da heurística é repetida até que o número total de soluções adicionadas em  $\Lambda$  ou  $\lambda$  alcance um limite. Esse limite é definido por  $l_c$  para soluções completas e  $l_p$  para parciais. A inclusão destas soluções em um dos conjuntos,  $\Lambda$  ou  $\lambda$ , ocorre durante a chamada da heurística construtiva.

---

**Algoritmo 1** Critério de parada para heurísticas construtivas

---

```

1: Entrada: uma heurística construtiva, número de soluções completas ( $l_c$ ) e parciais ( $l_p$ ) a ser obtida.
2: Saída:  $l_c$  soluções completas ou  $l_p$  soluções parciais, usando a heurística de entrada.
3: para cada depósito  $v \in V_d$  faça
4:   Número de iterações  $l_1 = l_2 = 0$ .
5:   enquanto  $l_1 < l_c$  and  $l_2 < l_p$  faça
6:     Inicializar  $|K|$  rotas vazias e fixar  $L_D$  como a lista de clientes não visitados (todos).
7:     Aplicar a heurística construtiva
8:     se uma solução completa é obtida então
9:        $l_1 \leftarrow l_1 + 1$ 
10:    senão
11:       $l_2 \leftarrow l_2 + 1$ 
12:    fim se
13:  fim enquanto
14: fim para

```

---

Cada heurística construtiva que compõem o algoritmo mateurístico é elencada e descrita em detalhes na sequência.

1. **Rotas aleatórias:** nesta heurística, as rotas são geradas sequencialmente, uma a uma. Cada rota inicia com a seleção aleatória de um cliente da lista de não visitados  $L_D$ , que é inserido ao final da mesma. Esse processo de inclusão de novos clientes se repete até que a rota se torne infactível, por falta de capacidade do veículo ou por extrapolar o horizonte de planejamento. Quando isso acontece, o último cliente inserido é removido, retorna à lista de não visitados e a rota é finalizada. Em seguida, caso ainda haja veículos disponíveis, uma nova rota inicia. Do contrário, se ainda houver clientes não designados, as rotas já finalizadas são adicionadas ao conjunto de soluções parciais  $\lambda$ . Caso a lista de não visitados esteja vazia, a solução completa gerada é adicionada ao conjunto  $\Lambda$ . O pseudocódigo relativo a esse procedimento é descrito no Algoritmo 2, e ele é repetido até que o número de soluções adicionadas a um dos conjuntos,  $\Lambda$  ou  $\lambda$ , alcance o limite estabelecido. Todas as soluções completas geradas também são

replicadas para os demais depósitos com o intuito de gerar diversidade. Da mesma forma, se a nova solução é completa, ela é adicionada ao conjunto  $\Lambda$ . Em caso contrário, o último cliente da rota infactível é removido e ela é testada novamente, repetindo o processo de remoção até que a mesma se torne factível. Após a reparação de todas as rotas infactíveis, a solução é adicionada ao conjunto  $\lambda$ . A replicação nos demais depósitos é descrita pelo Algoritmo 3.

---

**Algoritmo 2** Rotas aleatórias
 

---

```

1: Iniciar a primeira rota.
2: laço
3:   Selecionar aleatoriamente um cliente da lista  $L_D$ .
4:   Inserir o cliente selecionado ao final da rota atual.
5:   se a rota é factível então
6:     Remover o cliente da lista  $L_D$ .
7:     se  $L_D$  está vazia então
8:       Finalizar a rota e adicionar a solução completa ao conjunto  $\Lambda$ .
9:       Aplicar o Algoritmo 3 para replicar a solução aos demais depósitos.
10:    interromper laço
11:  fim se
12: senão
13:   Remover o último cliente da rota, retorná-lo à lista  $L_D$  e finalizar a rota.
14:   se há um veículo disponível então
15:     Iniciar uma nova rota.
16:   senão
17:     Adicionar a solução parcial ao conjunto  $\lambda$ .
18:   interromper laço
19: fim se
20: fim se
21: fim laço

```

---



---

**Algoritmo 3** Replicação nos demais depósitos
 

---

```

1: Entrada: Uma solução factível e completa associada ao depósito  $d$ .
2: Saída: Novas soluções, a partir da mesma sequência de clientes, mas associadas a outros depósitos.
3: para todo  $v \in V_d, v \neq d$  faça
4:   para cada rota da solução faça
5:     Trocar o depósito  $d$  pelo  $v$ .
6:     enquanto a nova rota é infactível faça
7:       Remover um cliente na ordem inversa da sua inserção.
8:     fim enquanto
9:   fim para
10:  se a nova solução é completa então
11:    Adicionar a solução ao conjunto  $\Lambda$ .
12:  senão
13:    Adicionar a solução ao conjunto  $\lambda$ .
14:  fim se
15: fim para

```

---

2. **Rotas aleatórias paralelas:** esta heurística tem uma dinâmica parecida com a anterior, mas impõe que todos os veículos disponíveis sejam necessariamente utilizados, iniciando a construção de todas as rotas simultaneamente. Os clientes são distribuídos sequencialmente aos veículos, através de um processo iterativo, descrito pelo Algoritmo 4. Quando não há mais clientes na lista de não visitados,  $L_D$ , todas as rotas são testadas. Caso alguma seja infactível, retiram-se aleatoriamente clientes dessa rota até que ela se torne factível. Todas elas são finalizadas quando não houver mais nenhuma infactível. A heurística é repetida até que um dos critérios de parada, descritos no Algoritmo 1, seja atendido. Como anteriormente, cada solução completa gerada é replicada para todos os outros depósitos, conforme o Algoritmo 3.

---

**Algoritmo 4** Rotas aleatórias paralelas
 

---

```

1: Iniciar  $|K|$  rotas.
2: enquanto  $L_D$  não está vazia faça
3:   para cada rota faça
4:     Selecionar aleatoriamente um cliente da lista  $L_D$ .
5:     Inserir o cliente selecionado ao final da rota atual.
6:     Remover o cliente da lista  $L_D$ .
7:     se  $L_D$  está vazia então
8:       interromper repetição
9:     fim se
10:   fim para
11: fim enquanto
12: se todas as rotas são factíveis então
13:   Finalizar as rotas e adicionar a solução completa ao conjunto  $\Lambda$ .
14:   Aplicar o Algoritmo 3 para replicar a solução aos demais depósitos.
15: senão
16:   enquanto uma rota é infactível faça
17:     Remover um cliente qualquer dessa rota
18:   fim enquanto
19:   Adicionar a solução parcial ao conjunto  $\lambda$ .
20: fim se

```

---

3.  **$\nu$ -vizinhos mais próximos:** nesta heurística, as rotas são geradas uma a uma, de forma sequencial. Para a designação do primeiro cliente de cada rota, são identificados os  $\nu$  clientes não visitados mais próximos do depósito, considerando o intervalo  $h = 0$ . Aqui, a noção de vizinho mais próximo é baseada no tempo necessário para deslocar-se até ele (atravessar o arco), e esse tempo varia conforme o intervalo  $h$  considerado. Após a identificação dos  $\nu$  vizinhos mais próximos, um desses clientes é selecionado randomicamente e adicionado à rota. Se a rota for factível, esse cliente é removido da lista de não visitados,  $L_D$ , e os tempos de deslocamento e de serviço nele são então computados. Isso permite identificar o momento em que o veículo está apto a seguir viagem e, consequentemente, o intervalo  $h$  para a nova seleção de vizinhos mais próximos. Caso a rota não seja factível, o último cliente adicionado é removido, retorna

à lista de não visitados,  $L_D$ , e a rota é finalizada. Esse processo de identificação dos  $v$  vizinhos mais próximos, seleção aleatória e inclusão em uma rota se repete até que todos os clientes tenham sido designados ou até que não haja mais veículos disponíveis. Novamente, se ao final todos os clientes estiverem designados a alguma rota, a solução gerada é adicionada ao conjunto  $\Lambda$ . Do contrário, a solução parcial é inserida no conjunto  $\lambda$ .

4.  **$v$ -vizinhos mais próximos – regressiva:** este procedimento é similar ao anterior, exceto pelo fato de a construção de cada rota inicia em ordem reversa, determinando a designação do último cliente em direção ao primeiro. Dessa forma, o intervalo de tempo considerado na primeira inserção de cada rota é o último intervalo  $h$  do horizonte de planejamento. Os vizinhos mais próximos são identificados com base no tempo necessário para deslocar-se deles em direção ao próximo local já designado, de acordo com o intervalo  $h$  correspondente ao momento da partida. O Algoritmo 5 apresenta o pseudocódigo para a heurística  $v$ -vizinhos mais próximos e também para sua versão reversa.
5.  **$v$ -vizinhos mais próximos – paralela:** esta heurística é similar a heurística  $v$ -vizinhos mais próximos, mas impõe que todos os veículos disponíveis sejam utilizados, iniciando a construção de todas as rotas simultaneamente. A distribuição de um cliente para cada veículo é sequencial. Após a inserção de todos os clientes, as rotas são testadas. Caso alguma seja infactível, retiram-se aleatoriamente clientes dessa rota até que ela se torne factível. Quando todas as rotas forem factíveis, elas são finalizadas. Obviamente, todos os veículos iniciam as suas rotas no intervalo  $h = 0$ , partindo do depósito em direção a um dos vizinhos mais próximos ainda não visitados. Novamente, se ao final todos os clientes estiverem designados a alguma rota, a solução completa gerada é adicionada ao conjunto  $\Lambda$ . Do contrário, a solução parcial é inserida no conjunto  $\lambda$ .
6.  **$v$ -vizinhos mais próximos – paralela e regressiva:** como na heurística  $v$ -vizinhos mais próximos – regressiva, aqui a construção das rotas inicia em ordem reversa, determinando a designação do último cliente em direção ao primeiro. Assim, o intervalo de tempo considerado na primeira inserção de cada rota é o último intervalo  $h$  do horizonte de planejamento. Além disso, todas as rotas iniciam simultaneamente, impondo a utilização de todos os veículos da frota. A distribuição de um cliente para cada veículo é sequencial. Da mesma forma, todas as rotas são testadas quando não há mais clientes na lista de não visitados. Caso alguma delas seja infactível, retiram-se aleatoriamente clientes dessa rota até que ela se torne factível. As rotas são finalizadas quando todas estiverem factíveis e então salvas em um dos conjuntos,  $\Lambda$  ou  $\lambda$ , conforme o tipo de solução gerada. O pseudocódigo para essas duas últimas heurísticas é apresentado no Algoritmo 6.



---

**Algoritmo 5**  $\nu$ -vizinhos mais próximos
 

---

```

1: se progressiva então
2:   Entrada: Clientes  $i \in V_c$  ordenados pelo menor tempo de deslocamento entre o depósito  $d$  e o
     cliente  $i$ , para  $h = 0$ . Para cada intervalo  $h \in H$  e para cada cliente  $i \in V_c$ , clientes  $j \in V_c, j \neq i$ 
     ordenados pelo menor tempo de deslocamento no arco  $(i, j)$ .
3: fim se
4: se regressiva então
5:   Entrada: Clientes  $i \in V_c$  ordenados pelo menor tempo de deslocamento entre o cliente  $i$  e o depósito
      $d$ , para  $h = m$ . Para cada intervalo  $h \in H$  e para cada cliente  $i \in V_c$ , clientes  $j \in V_c, j \neq i$  pelo
     menor tempo de deslocamento no arco  $(j, i)$ .
6: fim se
7: laço
8:   Selecionar aleatoriamente um dos  $\nu$  clientes mais próximos do último local inserido à rota (cliente
     ou depósito), considerando o intervalo corrente  $h$ .
9:   se progressiva então
10:    Inserir o cliente ao final da rota corrente.
11:   fim se
12:   se regressiva então
13:    Inserir o cliente no início da rota corrente, logo após o depósito.
14:   fim se
15:   se a rota é factível então
16:    Remover o cliente da lista  $L_D$ .
17:    se  $L_D$  está vazia então
18:      Finalizar a rota e adicionar a solução completa ao conjunto  $\Lambda$ .
19:      Aplicar o Algoritmo 3 para replicar a solução aos demais depósitos.
20:      interromper laço
21:    fim se
22:   senão
23:    Remover o cliente recém-adicionado à rota, retorná-lo à lista  $L_D$  e finalizar a rota.
24:    se há um veículo disponível então
25:      Iniciar uma nova rota.
26:    senão
27:      Adicionar a solução parcial ao conjunto  $\lambda$ .
28:      interromper laço
29:    fim se
30:   fim se
31: fim laço

```

---

7. **Enumeração:** através desta heurística, são criadas rotas ótimas com até  $\gamma$  clientes, usando a permutação dos mesmos para determinar a melhor sequência destes clientes. A motivação para tal procedimento está em gerar rotas curtas, mas com sequências ótimas, para adicionar ao conjunto de soluções parciais  $\lambda$ . O pseudocódigo desse procedimento é descrito pelo Algoritmo 7.

Para diversificar ainda mais a busca, gerando rotas relativamente mais curtas, executam-se as heurísticas construtivas 1–6 novamente, para um horizonte de planejamento menor. Para isso, reduz-se o número de intervalos  $h$ , mantendo-se, no entanto, a duração desses intervalos.

---

**Algoritmo 6**  $\nu$ -vizinhos mais próximos - paralela
 

---

```

1: se progressiva então
2:   Entrada: Clientes  $i \in V_c$  ordenados pelo tempo de deslocamento entre o depósito  $d$  e o cliente  $i$ ,
   para  $h = 0$ . Para cada intervalo  $h \in H$  e para cada cliente  $i \in V_c$ , clientes  $j \in V_c, j \neq i$  ordenados
   pelo tempo de deslocamento no arco  $(i, j)$ .
3: fim se
4: se regressiva então
5:   Entrada: Clientes  $i \in V_c$  ordenados pelo tempo de deslocamento entre o cliente  $i$  e o depósito  $d$ ,
   para  $h = m$ . Para cada intervalo  $h \in H$  e para cada cliente  $i \in V_c$ , clientes  $j \in V_c, j \neq i$  pelo tempo
   de deslocamento no arco  $(j, i)$ .
6: fim se
7: Iniciar  $|K|$  rotas.
8: enquanto  $L_D$  não está vazia faça
9:   para cada rota faça
10:    Selecionar aleatoriamente um dos  $\nu$  clientes mais próximos do local inserido à rota (cliente ou
    depósito), considerando o intervalo corrente  $h$ .
11:    se progressiva então
12:      Inserir o cliente ao final da rota corrente.
13:    fim se
14:    se regressiva então
15:      Inserir o cliente no início da rota corrente, logo após o depósito.
16:    fim se
17:    Remover o cliente da lista  $L_D$ .
18:    se  $L_D$  está vazia então
19:      interromper repetição
20:    fim se
21:  fim para
22: fim enquanto
23: se todas as rotas são factíveis então
24:   Finalizar as rotas e adicionar a solução completa ao conjunto  $\Lambda$ .
25:   Aplicar o Algoritmo 3 para replicar a solução aos demais depósitos.
26: senão
27:   enquanto uma rota é infactível faça
28:    Remover um cliente qualquer dessa rota
29:   fim enquanto
30:   Adicionar a solução parcial ao conjunto  $\lambda$ .
31: fim se

```

---

Todas as soluções obtidas, completas ou parciais, também são adicionadas, respectivamente, aos conjuntos  $\Lambda$  ou  $\lambda$ .

Além disso, quando sequências diferentes são geradas para o mesmo conjunto de vértices (depósito e clientes), calcula-se o custo de execução de cada uma delas, guardando apenas aquela de menor custo, que é aqui definida como rota não dominada. Assim, antes da inserção, todas as rotas adicionadas aos conjuntos  $\Lambda$  e  $\lambda$  são avaliadas, de modo que apenas rotas não dominadas sejam mantidas.

---

**Algoritmo 7** Enumeração de até  $\gamma$  clientes
 

---

```

1: para cada  $i \in V_d$  faça
2:   para  $\kappa = 1$  até  $\gamma$  faça
3:      $TempoTotalViagem \leftarrow Big - M$ 
4:     para cada possível combinação de  $\kappa$  clientes  $j \in V_c, (C_{|V_c|, \kappa})$  faça
5:       Inserir o depósito  $i$  no começo e no final de cada sequência de  $\kappa$  clientes permutados,  $(P_\kappa)$ 
6:       se essa rota é factível então
7:         Calcular o tempo total de deslocamento para execução da rota
8:         se o tempo total de deslocamento é menor que  $TempoTotalViagem$  então
9:            $TempoTotalViagem \leftarrow$  tempo total de deslocamento da rota atual
10:           $MelhorRota \leftarrow RotaAtual$ 
11:        fim se
12:      fim se
13:      Adicionar a melhor rota ao conjunto  $\lambda$ 
14:    fim para
15:  fim para
16: fim para

```

---

### 3.2.2 Modelo de particionamento de conjuntos

A última fase da mateurística busca combinar diferentes rotas dos conjuntos  $\Lambda$  e  $\lambda$ , oriundos da aplicação das heurísticas construtivas, com o intuito de obter melhores soluções para o problema. Para isso, utiliza-se um modelo de particionamento de conjuntos. A formulação matemática e os detalhes do problema são apresentados a seguir.

Sejam  $V_c$  um conjunto finito de  $m$  elementos (clientes),  $P_d = \{P_{d_1}, \dots, P_{d_n}\}$  o conjunto de todos os subconjuntos de  $V_c$  (clientes de uma rota) associados ao depósito  $d \in V_d$ , e  $R = \{1, \dots, n\}$  o conjunto de índices de  $P_d$  (índices de rotas). Um subconjunto  $F \subseteq R$  é uma partição de  $M$  se  $\cup_{i \in F} P_{d_i} = M$  e  $P_{d_i} \cap P_{d_j} = \emptyset \forall i, j \in F$ . O objetivo é encontrar a partição de menor custo associada a um depósito  $d \in V_d$ .

A cada rota  $r \in R$  está associado um parâmetro  $\Omega_r$ , que indica o depósito que está vinculado a ela, e um custo  $c_r$ , que representa o tempo acumulado para executar a rota, considerando o depósito como ponto de partida e de chegada. Define-se  $\alpha_{rj}$  como a designação do cliente  $j$  à rota  $r$ , isto é,  $\alpha_{rj} = 1$  se o cliente  $j$  é atendido pela rota  $r$ , e 0 em caso contrário.  $K$  é o conjunto de veículos disponível em cada depósito.

Dois conjuntos de variáveis são definidos. Para cada rota  $r \in R$ , a variável binária  $\chi_r$  assume valor 1 se a rota  $r$  é selecionada, e 0 caso contrário. Para cada depósito  $d \in V_d$ , a variável binária  $\Upsilon_d$  assume valor 1 se o depósito  $d$  é usado, e 0 caso contrário.

A formulação matemática é apresentada na sequência.

$$\min \sum_{i \in R} c_r \chi_r \quad (3.44)$$

sujeito a:

$$\sum_{r \in R} \alpha_{rj} \chi_r = 1, \quad \forall j \in V_c \quad (3.45)$$

$$\sum_{\substack{r \in R \\ \text{com } \Omega_r}} \chi_r \leq |K| \Upsilon_{(\Omega_r)} \quad (3.46)$$

$$\sum_{d \in V_d} \Upsilon_d = 1 \quad (3.47)$$

$$\chi_r \in \{0, 1\}, \quad \forall r \in R \quad (3.48)$$

$$\Upsilon_d \in \{0, 1\}, \quad \forall d \in V_d. \quad (3.49)$$

O objetivo (3.44) é encontrar um subconjunto de rotas vinculadas ao mesmo depósito, que atenda todos os clientes e seja executado no menor tempo possível. As restrições do conjunto (3.45) asseguram que cada cliente seja visitado exatamente uma vez. Restrições (3.46) impõem a conexão entre cada rota e seu respectivo depósito e garantem que o número de rotas executadas não exceda o número de veículos disponíveis. Através da restrição (3.47) é possível assegurar que todas as rotas da solução estão associadas ao mesmo depósito. Por fim, as restrições dos conjuntos (3.48) e (3.49) definem a natureza e o domínio das variáveis.

### 3.3 TESTES COMPUTACIONAIS

Nesta seção, especificam-se detalhes da configuração usada na condução dos testes computacionais, os parâmetros definidos para os algoritmos e a geração das instâncias de teste.

Os algoritmos foram implementados em linguagem C++. Usou-se o *solver* Gurobi Optimizer 8.1.1 para resolver os modelos de programação linear inteira mista, na configuração padrão. Todos os testes computacionais foram conduzidos em um processador Intel Core i7, rodando a 3.4 GHz, com 64 GB de memória RAM instalada, no sistema operacional Ubuntu Linux, com compilador gcc 7.4.0. Dois *threads* foram usados pelo *solver* e um limite total de tempo de 10800 segundos foi imposto na execução de cada configuração de teste.

Com o intuito de melhor ajustar o método, diferentes valores foram testados para cada um dos parâmetros que compõem as heurísticas construtivas. Os melhores resultados médios foram obtidos com os parâmetros fixados conforme especificado a seguir. Para todas as heurísticas, usou-se  $\nu = 2$ ,  $l_c = 5000$ , e  $l_p = 10000$ . Além disso, na repetição das heurísticas 1–6 para um horizonte de planejamento menor, reduziu-se o número de intervalos  $h$  em 1 e 2 unidades.

As dez melhores soluções completas obtidas pelas heurísticas construtivas foram armazenadas para servir como entrada (solução inicial) no modelo apresentado na Seção 3.1. Elas também forneceram estatísticas sobre as heurísticas responsáveis pela geração das melhores soluções completas.

### 3.3.1 Geração de instâncias

Para conduzir os testes computacionais, foram geradas aleatoriamente diversas classes de instâncias. Para criá-las, explorou-se a mesma base de dados apresentada em Belhassine et al. (2018). As informações geográficas disponíveis nessa base provêm de dados reais de tráfego da rede rodoviária da cidade de Québec – Canadá, gerados a partir de dados fornecidos por um parceiro industrial da área de varejo de móveis e eletrodomésticos. Adotou-se o LTM como método para fixar o tempo de deslocamento nos arcos, de acordo com os intervalos pré-estabelecidos. Como a base utilizada foi gerada a partir de informações reais de velocidade, ela respeita a propriedade FIFO, independentemente do método utilizado (LTM ou FSM). Devido ao aspecto temporal considerado e a flutuação da velocidade em diferentes vias, a desigualdade triangular não é respeitada. Considerando todas essas questões, foram definidos os parâmetros a seguir descritos.

Em um horizonte de planejamento de 15 horas, considerado entre 6h e 21h, definiram-se três intervalos de tempo de igual duração: 3600, 5400, e 10800 segundos. Assim, obteve-se instâncias compostas por 15, 10 ou 5 intervalos de tempo, que foram classificadas, respectivamente, em *grande*, *média* ou *pequena*.

Em cada instância considerou-se 1, 3, ou 5 potenciais depósitos e 10, 20, 50, 80 ou 100 clientes. A demanda (em unidades) de cada cliente foi gerada aleatoriamente a partir de uma distribuição uniforme discreta no intervalo  $[50, 750]$ , e o tempo de serviço (segundos), também de forma aleatória, com distribuição uniforme discreta no intervalo  $[1000, 10800]$ . Dessa forma, cada instância foi identificada pelo número de intervalos de tempo ( $|H|$ ), de depósitos ( $|V_d|$ ), de clientes ( $|V_c|$ ), de veículos ( $|K|$ ), e a capacidade de cada veículo ( $Q$ ). O número de veículos e a capacidade individual destes foram definidos a partir de testes preliminares.

Para cada possível combinação entre número de intervalos de tempo, de depósitos e de clientes, foram geradas aleatoriamente 5 instâncias. A Tabela 3.2 resume os parâmetros usados na criação dessas instâncias.

Conforme observou-se nos dados reais, obtidos da cidade de Québec e usados para geração das instâncias, o tempo de viagem em um arco pode ser inferior a um minuto, da ordem de segundos apenas. Isso porque há pequenas distâncias associadas a esses arcos, especialmente quando muitos vértices são considerados. Na Tabela 3.3 é possível visualizar algumas estatísticas dos tempos de viagem para instâncias com 3 potenciais depósitos. Em média, para todas elas, o tempo mínimo de deslocamento é inferior a um minuto, com valores médios não muito diferentes de 15 minutos. O tempo máximo de deslocamento nos arcos, em média, é inferior a uma hora. Isso condiz com o que foi definido como menor incremento de cada intervalo de tempo, na criação das instâncias.

Tabela 3.2: CONJUNTOS DE INSTÂNCIAS E SEUS PARÂMETROS

Tipo	$ H $	$\bar{T}$ (s)	$ V_d $	$ V_c $	$ K $	$Q$ (unid.)
Pequena	5	10800	{1, 3, 5}	10	3	4000
			{1, 3, 5}	20	4	4000
			{1, 3, 5}	50	8	4500
			{1, 3, 5}	80	12	4500
			{1, 3, 5}	100	15	4500
Média	10	5400	{1, 3, 5}	10	3	4000
			{1, 3, 5}	20	4	4000
			{1, 3, 5}	50	8	4500
			{1, 3, 5}	80	12	4500
			{1, 3, 5}	100	15	4500
Grande	15	3600	{1, 3, 5}	10	3	4000
			{1, 3, 5}	20	4	4000
			{1, 3, 5}	50	8	4500
			{1, 3, 5}	80	12	4500
			{1, 3, 5}	100	15	4500

FONTE: A autora (2019)

Tabela 3.3: INFORMAÇÕES SOBRE TEMPO DE DESLOCAMENTO PARA INSTÂNCIAS COM 3 DEPÓSITOS

Tipo	$ V_c $	Tempo de deslocamento (s)		
		Mínimo	Médio	Máximo
Pequena	10	132,40	1062,76	2684,00
	20	61,20	1087,72	2889,40
	50	27,80	1076,64	3009,60
	80	12,00	1074,56	3104,40
	100	15,00	1090,32	3275,20
Média	10	132,40	1089,34	3035,80
	20	61,20	1110,48	3041,40
	50	27,60	1102,20	3206,40
	80	12,00	1100,48	3241,00
	100	15,00	1117,08	3499,00
Grande	10	132,40	1090,32	2912,60
	20	61,20	1111,24	3111,20
	50	27,60	1101,52	3196,20
	80	12,00	1100,97	3311,40
	100	15,00	1117,15	3487,00
Média		49,65	1095,52	3133,64

FONTE: A autora (2019)

### 3.4 RESULTADOS E DISCUSSÃO

Nesta seção são apresentados e discutidos os resultados obtidos a partir de extensivos testes computacionais. Inicialmente, apresentam-se os resultados gerados a partir das heurísticas construtivas descritas na Seção 3.2.1. Em seguida, comparam-se os resultados da formulação matemática apresentada na Seção 3.1, considerando a inclusão, ou não, de um conjunto de soluções iniciais e de desigualdades válidas. Resultados detalhados da aplicação da metaheurística, descrita na Seção 3.2, são apresentados na sequência. Ao final, complementa-se esta seção

com uma comparação entre a formulação matemática proposta para o PLR-DT e um método sequencial, baseado na resolução de vários PRV-DT, um para cada potencial depósito.

### 3.4.1 Resultados das heurísticas construtivas

Reportam-se aqui os resultados gerados com a execução das heurísticas construtivas. A Tabela 3.4 apresenta os resultados das instâncias com 1 depósito, seguida pela Tabela 3.5, que refere-se aos conjuntos com 3 depósitos e, por fim, a Tabela 3.6 para instâncias com 5 depósitos. Elas reportam os resultados médios das 5 instâncias aleatórias criadas. As duas primeiras colunas de cada tabela contém informações sobre os parâmetros da instância. Na sequência, apresentam-se, respectivamente, a melhor solução obtida, tempo de execução (em segundos), número total de execuções das heurísticas, percentual de soluções completas e parciais, percentual de soluções criadas de forma única, total de rotas geradas e percentual de rotas únicas.

Tabela 3.4: RESULTADOS MÉDIOS DAS HEURÍSTICAS CONSTRUTIVAS PARA INSTÂNCIAS COM 1 DEPÓSITO

Tipo	$ V_c $	Melhor solução	Tempo (s)	Total de execuções	Soluções completas (%)	Soluções parciais (%)	Soluções únicas (%)	Rotas geradas	Rotas únicas (%)
Pequena	10	7840,80	14	124222,80	96,72	3,28	75,04	252081,40	80,99
	20	14501,00	39	141474,60	70,23	29,77	81,40	470320,00	75,99
	50	29890,80	208	187927,20	26,39	73,61	90,40	1346856,80	68,86
	80	44570,00	486	256332,20	10,54	89,46	93,34	2170857,20	72,09
	100	42332,00	703	340756,80	9,05	90,95	94,66	2786596,60	73,53
Média		27826,92	290,14	210142,72	42,59	57,41	86,97	1405342,40	74,29
Média	10	8138,20	13	120298,00	99,85	0,15	74,30	225542,60	84,98
	20	14650,80	32	135466,00	84,01	15,99	78,12	414935,00	76,37
	50	31356,00	204	183872,40	35,92	64,08	91,45	1245325,00	72,79
	80	45691,80	418	254282,20	17,94	82,06	95,68	2062134,40	77,81
	100	40861,20	623	326172,40	14,63	85,37	96,60	2443272,80	79,66
Média		28139,60	258,06	204018,20	50,47	49,53	87,23	1278241,96	78,32
Grande	10	7920,20	12	120295,00	99,85	0,15	74,39	226844,20	82,38
	20	14405,20	29	122577,60	98,01	1,99	74,36	331487,40	77,57
	50	29447,40	195	181759,80	50,48	49,52	87,41	1116946,60	73,26
	80	44363,40	476	242637,40	22,19	77,81	94,77	1857584,80	79,42
	100	42260,40	597	325596,60	15,27	84,73	96,26	2414417,80	79,18
Média		27679,32	261,87	198573,28	57,16	42,84	85,44	1189456,16	78,36
Média global		27881,95	270,02	204244,73	50,07	49,93	86,55	1291013,51	76,99

FONTE: A autora (2019)

A partir desses resultados, observa-se que o número de intervalos de tempo, identificado pelo tipo de instância (pequena, média, grande), não exerce efeito na complexidade do problema. Isso é indicado pela relação relativamente constante entre o número de rotas e o tempo necessário para gerá-las. Além disso, ao explorar mais de 10 milhões de rotas, (média para algumas configurações de instância), é perceptível que o conjunto de heurísticas construtivas criado é capaz de fornecer diversidade na sua geração. Ainda, é possível acrescentar que as médias alcançadas são coerentes com o fato de que o valor da função objetivo diminui quando o número

Tabela 3.5: RESULTADOS MÉDIOS DAS HEURÍSTICAS CONSTRUTIVAS PARA INSTÂNCIAS COM 3 DEPÓSITOS

Tipo	$ V_c $	Melhor solução	Tempo (s)	Total de execuções	Soluções completas (%)	Soluções parciais (%)	Soluções únicas (%)	Rotas geradas	Rotas únicas (%)
Pequena	10	7439,00	38	330800,60	89,22	10,78	89,94	807036,20	79,97
	20	13845,40	105	359439,60	62,58	37,42	91,43	1379741,00	69,96
	50	26563,40	533	455290,20	27,81	72,19	94,54	3278719,40	63,63
	80	37442,80	1459	666719,00	12,36	87,64	96,64	5322420,80	65,85
	100	46378,20	1982	902420,40	6,02	93,98	96,71	6687301,60	65,32
Média		26333,76	823,56	542933,96	39,60	60,40	93,85	3495043,80	68,95
Média	10	7491,80	36	301164,80	98,99	1,01	89,22	689579,40	82,09
	20	13825,00	82	312543,00	94,90	5,10	88,46	1022341,00	75,29
	50	26747,60	573	446096,60	42,40	57,60	94,99	3003849,20	69,34
	80	39722,40	1110	635934,60	12,73	87,27	98,50	4934900,20	70,57
	100	46120,40	1814	888654,60	12,87	87,13	98,56	6264464,20	74,13
Média		26781,44	723,05	516878,72	52,38	47,62	93,95	3183026,80	74,29
Grande	10	7467,40	34	300765,00	99,19	0,81	89,07	677091,20	81,32
	20	13929,00	82	307261,60	96,60	3,40	88,20	1002590,60	75,40
	50	26500,40	555	452538,80	47,45	52,55	94,43	2960578,60	69,87
	80	39293,60	1070	626789,80	21,77	78,23	97,49	4583832,20	74,34
	100	45920,20	1711	877078,40	16,61	83,39	97,94	5890240,00	75,96
Média		26622,12	690,44	512886,72	56,32	43,68	93,42	3022866,52	75,38
Média global		26579,11	745,68	524233,13	49,43	50,57	93,74	3233645,71	72,87

FONTE: A autora (2019)

Tabela 3.6: RESULTADOS MÉDIOS DAS HEURÍSTICAS CONSTRUTIVAS PARA INSTÂNCIAS COM 5 DEPÓSITOS

Tipo	$ V_c $	Melhor solução	Tempo (s)	Total de execuções	Soluções completas (%)	Soluções parciais (%)	Soluções únicas (%)	Rotas geradas	Rotas únicas (%)
Pequena	10	7929,40	63	525937,40	88,73	11,27	93,21	1346014,00	78,70
	20	12809,20	185	596397,40	61,95	38,05	94,23	2349380,20	70,56
	50	26219,80	938	739134,00	24,67	75,33	95,96	5418796,20	60,35
	80	35662,60	2568	1060159,40	12,14	87,86	97,50	8272246,60	64,80
	100	39375,60	3157	1465692,40	6,74	93,26	97,58	10561756,20	65,05
Média		24399,32	1382,30	877464,12	38,84	61,16	95,70	5589638,64	67,89
Média	10	7866,60	56	481350,00	99,17	0,83	92,67	1140974,40	80,62
	20	13111,00	141	523431,80	88,76	11,24	92,93	1861693,60	75,12
	50	26624,40	918	709032,40	41,43	58,57	96,61	4833388,00	67,92
	80	36179,60	1804	1057069,60	19,80	80,20	98,39	8005109,20	73,00
	100	39321,40	2938	1452603,60	11,67	88,33	98,85	10115220,00	73,55
Média		24620,60	1171,34	844697,48	52,17	47,83	95,89	5191277,04	74,04
Grande	10	7479,80	56	481235,00	99,12	0,88	92,79	1117644,00	79,89
	20	13018,80	138	500488,60	94,67	5,33	92,50	1738327,40	76,02
	50	26897,60	905	697168,00	36,58	63,42	96,83	4789906,40	66,49
	80	37212,60	1666	1005103,80	20,43	79,57	98,41	7318312,40	73,50
	100	38906,60	2582	1413470,80	14,19	85,81	98,69	9401377,60	74,75
Média		24703,08	1069,19	819493,24	53,00	47,00	95,84	4873113,56	74,13
Média global		24574,33	1207,61	847218,28	48,00	52,00	95,81	5218009,75	72,02

FONTE: A autora (2019)

de depósitos aumenta, pois a instância base é a mesma, mas novos locais são adicionados para cada um dos potenciais depósitos inseridos. Assim, espera-se que o valor da função objetivo



diminua frente a um problema de otimização mais difícil. Por fim, o tempo médio de execução é inferior a 30 minutos, o que é aceitável, considerando a complexidade do problema.

### 3.4.2 Resultados da formulação matemática

Na sequência, são apresentados os resultados alcançados com o modelo matemático proposto para o PLR-DT, definido por (3.1) – (3.24). Eles traduzem a média obtida com a execução das 5 instâncias criadas para cada configuração, em um tempo limite de 10800 segundos. Inicialmente, reportam-se os resultados gerados quando o conjunto de soluções iniciais é fornecido como entrada, associado com a inclusão das desigualdades válidas. A Tabela 3.7 resume os resultados obtidos para 1 depósito, enquanto a Tabela 3.8 refere-se aos conjuntos com 3 depósitos e, por fim, a Tabela 3.9 trata instâncias com 5 depósitos. Assim como nas tabelas anteriores, as duas primeiras colunas fornecem informações sobre as instâncias. Para comparação entre o desempenho dos diferentes métodos, cada tabela traz novamente a melhor solução produzida pelas heurísticas construtivas (HC Melhor solução). Na sequência, apresentam-se a melhor solução obtida com o modelo, ou limite superior (LS), o limite inferior (LI), o *gap* entre esses limites (em %), calculado como  $100(LS - LI)/LS$ , o tempo de execução (em segundos) e a melhoria alcançada (em %), quando comparada com a solução das heurísticas construtivas.

Tabela 3.7: RESULTADOS MÉDIOS DO PLR-DT COM DESIGUALDADES VÁLIDAS E SOLUÇÕES INICIAIS PARA INSTÂNCIAS COM 1 DEPÓSITO

Tipo	$ V_c $	HC	PLR-DT				Melhoria (%)
		Melhor solução	Melhor solução	Limite Inferior	Gap (%)	Tempo (s)	
Pequena	10	7840,80	7595,40	7595,40	0,00	23	3,21
	20	14501,00	13434,60	11080,47	17,91	10800	6,86
	50	29890,80	29252,40	16279,08	43,62	10800	2,25
	80	44570,00	43815,40	22330,00	48,54	10800	1,77
	100	42332,00	41457,60	20511,04	50,06	10800	2,10
Média		27826,92	27111,08	15559,20	32,03	8644,70	3,24
Média	10	8138,20	7923,20	7923,20	0,00	320	2,46
	20	14650,80	13850,20	11140,31	19,14	10800	5,61
	50	31356,00	30970,40	16164,96	47,28	10800	1,17
	80	45691,80	45470,40	22816,70	49,46	10800	0,48
	100	40861,20	40728,40	21084,42	47,86	10800	0,48
Média		28139,60	27788,52	15825,92	32,75	8703,96	2,04
Grande	10	7920,20	7803,00	7803,00	0,00	128	1,44
	20	14405,20	13736,80	11050,30	19,52	10800	4,21
	50	29447,40	29164,80	16180,34	43,86	10800	0,93
	80	44363,40	44263,20	22682,70	48,35	10800	0,24
	100	42260,40	42009,20	20704,38	50,25	10800	0,55
Média		27679,32	27395,40	15684,14	32,40	8665,80	1,47
Média global		27881,95	27431,67	15689,75	32,39	8671,49	2,25

FONTE: A autora (2019)

Tabela 3.8: RESULTADOS MÉDIOS DO PLR-DT COM DESIGUALDADES VÁLIDAS E SOLUÇÕES INICIAIS PARA INSTÂNCIAS COM 3 DEPÓSITOS

Tipo	$ V_c $	HC	PLR-DT			Melhoria (%)
		Melhor solução	Melhor solução	Limite Inferior	Gap (%)	Tempo (s)
Pequena	10	7439,00	7030,40	7030,40	0,00	99
	20	13845,40	13210,60	10729,50	18,34	10800
	50	26563,40	26111,80	15943,84	38,70	10800
	80	37442,80	37103,20	19964,80	45,81	10800
	100	46378,20	45999,80	24046,12	47,59	10800
	Média	26333,76	25891,16	15542,93	30,09	8659,82
Média	10	7491,80	7252,60	7252,60	0,00	378
	20	13825,00	13439,40	10588,76	21,03	10800
	50	26747,60	26455,60	16059,90	39,23	10800
	80	39722,40	39535,80	20253,78	48,50	10800
	100	46120,40	45740,80	24177,66	47,13	10800
	Média	26781,44	26484,84	15666,54	31,18	8715,68
Grande	10	7467,40	7153,00	7153,00	0,00	221
	20	13929,00	13617,00	10637,12	21,86	10800
	50	26500,40	26316,20	15968,34	39,29	10800
	80	39293,60	39138,00	20259,94	47,84	10800
	100	45920,20	45885,00	23987,42	47,55	10800
	Média	26622,12	26421,84	15601,16	31,31	8684,34
Média global		26579,11	26265,95	15603,55	30,86	8686,61

FONTE: A autora (2019)

Essas tabelas mostram que, mesmo com a presença de todas as desigualdades válidas e com a inclusão de soluções iniciais, o *gap* ainda continua elevado, após 3 horas de execução, sugerindo que este problema permanece sendo de difícil resolução. Além disso, o *solver* Gurobi não foi capaz de melhorar significativamente as soluções iniciais fornecidas, gerando uma melhoria média de apenas 2,25% para instâncias com 1 depósito, 1,96% para casos com 3 depósitos, e 1,68% para instâncias com 5 depósitos. Embora em instâncias com um número maior de clientes as melhorias sejam apenas marginais, a inclusão de soluções iniciais mostrou-se realmente eficiente, conforme será confirmado pelos resultados que serão apresentados na sequência. Sem esse conjunto, o *solver* não foi capaz de encontrar nenhuma solução factível.

A seguir, é realizada uma análise acerca da eficiência da utilização do conjunto de soluções iniciais e da importância da inclusão das desigualdades válidas nesse contexto. Inicialmente, analisa-se a situação em que o conjunto de soluções iniciais não é fornecido, mas mantêm-se todas as desigualdades válidas presentes no modelo. A Tabela 3.10 concentra resultados médios obtidos com a formulação matemática do PLR-DT e com a inclusão das desigualdades válidas. Novamente, as duas primeiras colunas trazem informações a respeito da instância. Na sequência, para cada configuração relativa ao número de depósitos, têm-se os resultados da melhor solução obtida, ou limite superior (LS), o limite inferior, e seus respectivos *gaps*. Estes últimos traduzem a deterioração das soluções em comparação aquelas mostradas nas Tabelas 3.7 – 3.9. Esse *gap* é calculado, no caso do limite superior, como  $100(LS_c - LS_a)/LS_c$ ,

Tabela 3.9: RESULTADOS MÉDIOS DO PLR-DT COM DESIGUALDADES VÁLIDAS E SOLUÇÕES INICIAIS PARA INSTÂNCIAS COM 5 DEPÓSITOS

Tipo	$ V_c $	HC	PLR-DT				Melhoria (%)
		Melhor solução	Melhor solução	Limite Inferior	Gap (%)	Tempo (s)	
Pequena	10	7929,40	7604,00	7604,00	0,00	664	3,89
	20	12809,20	12278,00	9969,72	18,69	10800	4,24
	50	26219,80	25346,00	15660,88	38,20	10800	3,34
	80	35662,60	35229,60	19472,56	44,62	10800	1,23
	100	39375,60	39085,40	19794,28	49,51	10800	0,66
	Média	24399,32	23908,60	14500,29	30,20	8772,83	2,67
Média	10	7866,60	7790,40	7540,06	2,38	2357	1,28
	20	13111,00	12830,60	10202,14	20,20	10800	2,19
	50	26624,40	26411,60	15669,18	40,65	10800	0,80
	80	36179,60	35914,00	19714,08	45,04	10800	0,70
	100	39321,40	39207,20	20213,74	48,66	10800	0,29
	Média	24620,60	24430,76	14667,84	31,39	9111,37	1,05
Grande	10	7479,80	7355,40	7355,40	0,00	904	1,61
	20	13018,80	12655,00	10068,05	20,52	10800	2,84
	50	26897,60	26644,80	15644,86	41,29	10800	0,93
	80	37212,60	36945,80	19494,56	47,22	10800	0,67
	100	38906,60	38718,60	19895,56	48,86	10800	0,50
	Média	24703,08	24463,92	14491,69	31,58	8820,89	1,31
Média global		24574,33	24267,76	14553,27	31,06	8901,70	1,68

FONTE: A autora (2019)

onde  $LS_c$  indica o limite superior da formulação completa apresentada nas Tabelas 3.7 – 3.9 e  $LS_a$  indica o respectivo limite superior da formulação atual. O mesmo procedimento é aplicado ao limite inferior. O asterisco indica que nenhuma solução foi obtida nas cinco instâncias testadas.

Os resultados na Tabela 3.10 mostram que, para instâncias com um número maior de clientes, nenhuma solução factível foi encontrada. As colunas associadas aos *gaps* indicam que em algumas instâncias para as quais uma solução factível é obtida, sua qualidade é significativamente pior do que a fornecida pelas heurísticas construtivas (mais de 4% pior para instâncias com 3 depósitos). Em relação aos limites inferiores, na maioria dos casos eles são equivalentes, apresentando uma pequena piora (em torno de 0,5%) em relação à situação anterior. É perceptível que, em média, o limite superior piora e que fornecer um conjunto de soluções iniciais impacta positivamente para que o *solver* consiga encontrar outras soluções. Na maioria dos casos, ele sozinho não foi capaz de gerar solução factível, mesmo após três horas de execução.

Analisando resultados mais detalhados, verificou-se ainda que em instâncias com poucos clientes, o tempo necessário para provar a otimalidade diminuiu 27,75% quando o conjunto de soluções iniciais foi fornecido como entrada. Dessa forma, fica evidente que os resultados mostrados nas Tabelas 3.7 – 3.9 são fortemente dependentes da qualidade das soluções fornecidas.

Na Tabela 3.11 apresentam-se os resultados obtidos com a formulação matemática e a inclusão de um conjunto de soluções iniciais sem, no entanto, utilizar as desigualdades válidas. Novamente, os resultados são comparados com as soluções reportadas pelas Tabelas 3.7 – 3.9.

Tabela 3.10: RESULTADOS MÉDIOS DO PLR-DT SEM SOLUÇÕES INICIAIS E COM DESIGUALDADES VÁLIDAS

Tipo	$ V_c $	1 depósito				3 depósitos				5 depósitos			
		Melhor Solução	Limite Inferior	LS gap (%)	LI gap (%)	Melhor Solução	Limite Inferior	LS gap (%)	LI gap (%)	Melhor Solução	Limite Inferior	LS gap (%)	LI gap (%)
Pequena	10	7595,40	7595,40	0,00	0,00	7030,40	7030,40	0,00	0,00	7604,00	7604,00	0,00	0,00
	20	13407,00	11074,81	-0,26	0,02	13097,60	10656,36	-0,82	-0,72	12402,60	9882,85	1,14	-0,91
	50	*	16046,50	*	-1,50	27061,00†	15816,64	13,57†	-0,81	*	15453,46	*	-1,34
	80	*	22276,07	*	-0,24	*	19893,47	*	-0,35	*	19445,88	*	-0,13
	100	*	20455,48	*	-0,25	*	23971,18	*	-0,31	*	19824,53	*	0,14
	Média	*	15489,65	-0,13	-0,39	*	15473,61	4,25	-0,44	*	14442,14	0,57	-0,45
Média	10	7923,20	7923,20	0,00	0,00	7252,60	7252,60	0,00	0,00	7788,40	7497,55	-0,02	-0,46
	20	13910,80	10957,18	0,67	-1,57	14245,60	10419,60	5,43	-1,67	14737,20	9914,20	12,65	-2,84
	50	*	15992,20	*	-1,10	*	15920,85	*	-0,87	*	15594,41	*	-0,48
	80	*	22727,76	*	-0,41	*	20250,26	*	-0,02	*	19716,04	*	0,02
	100	*	21094,17	*	0,05	*	24159,95	*	-0,08	*	20217,56	*	0,05
	Média	*	15738,90	0,33	-0,61	*	15600,65	2,72	-0,53	*	14587,95	6,32	-0,74
Grande	10	7803,00	7803,00	0,00	0,00	7153,00	7153,00	0,00	0,00	7355,40	7355,40	0,00	0,00
	20	15016,20	10857,30	8,43	-1,73	15672,20	10328,77	12,66	-3,09	14007,60	9893,00	9,22	-1,72
	50	*	16083,83	*	-0,60	*	15923,53	*	-0,28	*	15567,25	*	-0,48
	80	*	22730,60	*	0,22	*	20221,82	*	-0,18	*	19535,39	*	0,21
	100	*	20733,15	*	0,09	*	23983,37	*	-0,03	*	19878,79	*	-0,07
	Média	*	15641,58	4,21	-0,40	*	15522,10	6,33	-0,71	*	14445,97	4,61	-0,41
Média global		*	15623,38	1,47	-0,47	*	15532,12	4,43	-0,56	*	14492,02	3,83	-0,54

† Em três instâncias (de cinco) não foi encontrada solução factível. O *gap* foi calculado com base nas soluções das duas instâncias restantes.

FONTE: A autora (2019)

Tabela 3.11: RESULTADOS MÉDIOS DO PLR-DT COM SOLUÇÕES INICIAIS E SEM DESIGUALDADES VÁLIDAS

Tipo	$ V_c $	1 depósito				3 depósitos				5 depósitos			
		Melhor Solução	Limite Inferior	LS gap (%)	LI gap (%)	Melhor Solução	Limite Inferior	LS gap (%)	LI gap (%)	Melhor Solução	Limite Inferior	LS gap (%)	LI gap (%)
Pequena	10	7595,40	7595,40	0,00	0,00	7030,40	7030,40	0,00	0,00	7604,00	7604,00	0,00	0,00
	20	13311,20	9988,23	-1,23	-10,42	13227,20	10076,76	0,23	-6,51	12307,20	9288,06	0,13	-7,57
	50	28871,20	14583,04	-1,31	-11,70	26046,20	14447,74	-0,27	-10,35	25313,60	14394,86	-0,14	-8,81
	80	43716,20	18702,36	-0,19	-20,79	36735,80	16693,08	-1,07	-19,55	34855,00	17047,44	-1,11	-14,18
	100	41356,60	17552,94	-0,16	-17,19	45129,80	19902,24	-1,98	-20,91	38968,20	17696,96	-0,26	-11,44
Média		26970,12	13684,39	-0,58	-12,02	25633,88	13630,04	-0,62	-11,46	23809,60	13206,26	-0,27	-8,40
Média	10	7923,20	7923,20	0,00	0,00	7252,60	7252,60	0,00	0,00	7788,40	7420,80	-0,02	-1,37
	20	13774,60	9891,14	-0,25	-12,40	13182,60	10096,44	-1,91	-4,96	12835,80	9254,37	0,07	-10,41
	50	30832,60	14374,22	-0,47	-12,60	26233,80	14373,76	-0,85	-11,75	26030,60	14249,62	-1,47	-10,00
	80	45103,60	18519,72	-0,79	-24,80	39337,00	16535,90	-0,46	-22,50	35712,60	16953,24	-0,59	-16,25
	100	40646,00	17411,68	-0,34	-21,19	45509,00	19938,82	-0,61	-21,32	39065,40	17632,56	-0,33	-14,35
Média		27656,00	13623,99	-0,37	-14,20	26303,00	13639,50	-0,77	-12,11	24286,56	13102,12	-0,47	-10,48
Grande	10	7803,00	7803,00	0,00	0,00	7153,00	7153,00	0,00	0,00	7355,40	7355,40	0,00	0,00
	20	13825,00	9564,46	0,43	-15,70	13605,00	9873,72	-0,08	-7,80	12811,00	9152,32	1,23	-10,22
	50	29195,60	14294,08	0,12	-13,28	26282,20	14261,26	-0,14	-11,98	26672,20	14158,56	0,08	-10,54
	80	44046,80	18301,42	-0,50	-25,39	39155,40	16479,72	0,03	-22,91	37044,40	16781,00	0,19	-16,12
	100	41854,60	17305,12	-0,46	-19,95	45701,80	19682,70	-0,40	-21,87	38480,60	17413,48	-0,84	-13,78
Média		27345,00	13453,62	-0,08	-14,86	26379,48	13490,08	-0,12	-12,91	24472,72	12972,15	0,13	-10,13
Média global		27323,71	13587,33	-0,34	-13,69	26105,45	13586,54	-0,50	-12,16	24189,63	13093,51	-0,20	-9,67

FONTE: A autora (2019)

Comparando com os resultados mostrados na Tabela 3.10, nota-se que ao fornecer uma solução inicial garante-se a obtenção de um limite superior para todas as instâncias, mesmo aquelas com um número maior de clientes. Com a inexistência das desigualdades válidas, o modelo apresenta menos restrições. Isso produz qualidade de solução basicamente equivalente (soluções marginalmente melhores, com uma melhoria de 0,34% para instâncias com 1 depósito, de 0,50% para casos com 3 depósitos e 0,20% para instâncias com 5 depósitos). Além disso, conforme esperado, a inclusão de desigualdades válidas melhora consideravelmente os limites inferiores. Isso pode ser visto através dos resultados apresentados nas colunas *LI gap (%)* da Tabela 3.11. Em relação aos limites inferiores, quando comparam-se os valores das Tabelas 3.7 – 3.9 com os resultados dos testes sem a inclusão das desigualdades válidas, nota-se que eles pioram drasticamente (valores muito menores). Embora onipresente, esse efeito fica mais evidente em instâncias com um número maior de clientes.

Esses resultados ajudam a destacar a importância de dois aspectos desenvolvidos neste trabalho: as desigualdades válidas, para melhorar os limites inferiores, e as heurísticas construtivas, para ajudar a fornecer soluções factíveis, especialmente em instâncias maiores.

### 3.4.3 Resultados da abordagem mateurística

As Tabelas 3.12 – 3.14 mostram os resultados médios gerados pela abordagem mateurística proposta. Dispõe-se uma tabela para cada diferente conjunto de instâncias, no que diz respeito ao número de depósitos. As duas primeiras colunas contêm informações sobre as instâncias. Os resultados obtidos com as heurísticas construtivas já foram apresentados na Seção 3.4.1, mas para facilitar a análise, eles são apresentados novamente aqui, da terceira à quarta coluna. Na sequência, expõem-se os resultados médios alcançados com as 5 instâncias criadas em cada configuração, após a aplicação do problema de particionamento de conjuntos. Descreve-se o número de rotas não dominadas dessa fase, a solução obtida e o tempo de execução (em segundos). Por fim, as duas últimas colunas informam, respectivamente, a melhoria alcançada em relação à solução das heurísticas construtivas e sobre os resultados do modelo proposto para o PLR-DT, com soluções iniciais e desigualdades válidas, apresentados nas Tabelas 3.7 – 3.9.

Diversas observações interessantes podem ser extraídas da análise dos resultados apresentados nas Tabelas 3.12 – 3.14. Primeiro, observa-se que as heurísticas construtivas exploram muito bem o sequenciamento de clientes. De um grande número de rotas geradas, muitas são permutações dos mesmos clientes, resultando em um número bem menor de rotas não dominadas. Esse tratamento tem a vantagem de reduzir a carga computacional relacionada à resolução do modelo de particionamento de conjuntos. Essas tabelas mostram também que a aplicação do particionamento definido melhora ainda mais as soluções geradas pelas heurísticas. A melhoria em relação a estas é de 8,99% para instâncias com 1 depósito, 9,18% para instâncias com 3 depósitos e 9,76% para instâncias com 5 depósitos. Além disso, o tempo médio de execução na abordagem mateurística é menor que o tempo relatado nas Tabelas 3.7 – 3.9, o que pode ainda ser insignificante se comparado ao impacto que essas soluções teriam na prática.

Tabela 3.12: RESULTADOS MÉDIOS DA MATEURÍSTICA PARA INSTÂNCIAS COM 1 DEPÓSITO

Tipo	$ V_c $	Heurísticas Construtivas		Problema de Particionamento de Conjuntos				
		Melhor solução	Tempo (s)	Rotas não dominadas	Solução	Tempo (s)	Melhoria sobre	
							HC (%)	PLR-DT (%)
Pequena	10	7840,80	14	975	7828,20	<1	0,20	-3,17
	20	14501,00	39	44024	13574,40	1	6,31	-0,71
	50	29890,80	208	556242	26936,40	13	9,99	7,88
	80	44570,00	486	1107071	38470,80	51	13,63	12,02
	100	42332,00	703	1537024	35867,80	88	15,15	13,33
Média		27826,92	290	649067,08	24535,52	31	9,05	5,87
Média	10	8138,20	13	963	8057,80	< 1	0,97	-1,55
	20	14650,80	32	45576	13822,80	1	5,44	-0,29
	50	31356,00	204	661500	27072,20	26	13,63	12,60
	80	45691,80	418	1240663	39115,20	55	14,46	14,04
	100	40861,20	623	1623787	35373,80	191	13,58	13,16
Média		28139,60	258	714497,84	24688,36	55	9,62	7,59
Grande	10	7920,20	12	791	7859,60	< 1	0,64	-0,81
	20	14405,20	29	51423	13932,80	1	3,16	-1,21
	50	29447,40	195	616601	26121,60	23	11,15	10,32
	80	44363,40	476	1198068	39041,80	152	12,12	11,91
	100	42260,40	597	1601521	36226,20	78	14,38	13,91
Média		27679,32	262	693681	24636,40	51	8,29	6,83
Média global		27881,95	270	685749	24620,09	45	8,99	6,76

FONTE: A autora (2019)

A Tabela 3.12 resume os resultados para instâncias com 1 depósito. Conforme mostrado nela, a aplicação da fase particionamento de conjuntos melhora a solução média em todo conjunto de instâncias. A melhoria é maior nos conjuntos com mais de 50 clientes. A combinação de diferentes rotas pode proporcionar maior qualidade, melhorando, em alguns casos, em mais de 13% os resultados obtidos com a formulação matemática. Inclusive, às vezes essa abordagem possibilita a obtenção da solução ótima, conforme pôde ser comprovado através da análise de resultados mais detalhados, gerados pelo modelo exato.

Da mesma forma, a Tabela 3.13 mostra que a aplicação da fase de particionamento pode melhorar a solução encontrada pelas heurísticas construtivas em todas as instâncias. Melhores resultados médios são obtidos para configurações com mais de 50 clientes. Em instâncias com 5 depósitos, a fase de otimização obtém a maior melhoria média global (8,14%), como pode ser visto na Tabela 3.14. Em instâncias individuais observa-se uma melhoria média superior a 15%, destacando a importância de combinar diferentes soluções de forma inteligente.

#### 3.4.4 Comparação entre a formulação matemática e uma abordagem sequencial

Compara-se aqui a formulação matemática proposta para o PLR-DT com uma abordagem sequencial, baseada na resolução de vários PRV-DT, um para cada potencial depósito. Para resolver o PRV-DT, usou-se uma formulação matemática baseada no modelo proposto para o

Tabela 3.13: RESULTADOS MÉDIOS DA MATEURÍSTICA PARA INSTÂNCIAS COM 3 DEPÓSITOS

Tipo	$ V_c $	Heurísticas Construtivas		Problema de Particionamento de Conjuntos				
		Melhor solução	Tempo (s)	Rotas não dominadas	Solução	Tempo (s)	Melhoria sobre	
							HC (%)	PLR-DT (%)
Pequena	10	7439,00	38	2811	7436,00	0	0,04	-5,73
	20	13845,40	105	127821	13182,40	8	4,75	0,06
	50	26563,40	533	1231663	23624,20	206	11,09	9,50
	80	37442,80	1459	2328382	33212,80	1251	11,30	10,51
	100	46378,20	1982	3027884	39759,80	3480	14,53	13,82
Média		26333,76	824	1343712,28	23443,04	989	8,34	5,63
Média	10	7491,80	36	2568	7413,40	0	1,05	-2,22
	20	13825,00	82	158806	13248,40	10	4,09	1,33
	50	26747,60	573	1428111	23451,60	310	12,42	11,43
	80	39722,40	1110	2476276	34467,20	1402	13,29	12,89
	100	46120,40	1814	3645606	38704,20	2961	16,16	15,52
Média		26781,44	723	1542273,48	23456,96	937	9,40	7,79
Grande	10	7467,40	34	2381	7378,00	0	1,17	-3,14
	20	13929,00	82	156217	13328,80	6	4,27	2,14
	50	26500,40	555	1487955	23261,60	289	12,17	11,54
	80	39293,60	1070	2640048	33240,80	1292	15,47	15,14
	100	45920,20	1711	3669709	38691,20	2919	15,83	15,76
Média		26622,12	690	1591262	23180,08	901	9,78	8,29
Média global		26579,11	746	1492416	23360,03	942	9,18	7,24

FONTE: A autora (2019)

PLR-DT, e definiram-se as instâncias para cada um dos potenciais depósitos. Para adaptar o modelo, excluíram-se as variáveis binárias  $w_i, \forall i \in V_d$ , uma vez que o PRV-DT considera um único depósito, já aberto. Também alteraram-se as restrições (3.4), (3.5) e (3.12), para que a variável  $w_i$  seja substituída por 1. Além disso, eliminou-se o conjunto (3.6). Todas as demais restrições e desigualdades válidas foram mantidas, conforme o modelo do PLR-DT. A formulação matemática usada para o PRV-DT é apresentada no Apêndice A. Os métodos heurísticos para geração da solução inicial puderam ser usados sem nenhuma modificação.

Esses testes computacionais consistiram em executar os dois métodos em um subconjunto de instâncias, até a otimalidade. Executou-se o modelo para o PLR-DT pelo tempo necessário para alcançar a solução ótima e, da mesma forma, isso também foi feito para o PRV-DT, em cada potencial depósito. Em seguida, comparou-se o tempo de execução do PLR-DT com o tempo total necessário para executar todos os PRV-DT correspondentes. Esse procedimento foi realizado para todas as instâncias com 10 clientes e com 3 ou 5 depósitos. As Tabelas 3.15 e 3.16 apresentam esses resultados.

Os resultados da Tabela 3.15 mostram claramente que a abordagem integrada do PLR-DT supera significativamente a do PRV-DT sequencial. Observa-se que o PLR-DT é capaz de encontrar uma solução ótima em um tempo significativamente menor, para todas as instâncias



Tabela 3.14: RESULTADOS MÉDIOS DA MATEURÍSTICA PARA INSTÂNCIAS COM 5 DEPÓSITOS

Tipo	V <sub>c</sub>	Heurísticas Construtivas		Problema de Particionamento de Conjuntos				
		Melhor solução	Tempo (s)	Rotas não dominadas	Solução	Tempo (s)	Melhoria sobre	
							HC (%)	PLR-DT (%)
Pequena	10	7929,40	63	4589	7767,80	0	1,85	-2,11
	20	12809,20	185	211555	12073,20	16	5,68	1,45
	50	26219,80	938	1787269	23107,40	415	11,91	8,86
	80	35662,60	2568	3380900	30723,00	1601	13,86	12,78
	100	39375,60	3157	4728847	33290,60	6776	15,12	14,56
Média		24399,32	1382	2022631,88	21392,40	1762	9,69	7,11
Média	10	7866,60	56	4422	7833,00	0	0,57	-0,76
	20	13111,00	141	242627	12609,20	19	3,87	1,67
	50	26624,40	918	2188427	22985,40	625	13,69	12,98
	80	36179,60	1804	4114079	30604,20	1278	15,34	14,75
	100	39321,40	2938	5658738	33065,00	6427	15,62	15,37
Média		24620,60	1171	2441658,60	21419,36	1670	9,82	8,80
Grande	10	7479,80	56	4404	7479,80	0	0,00	-1,66
	20	13018,80	138	248305	12376,60	19	4,95	2,12
	50	26897,60	905	2102291	23406,00	599	13,01	12,19
	80	37212,60	1666	3922798	31151,60	2154	16,26	15,69
	100	38906,60	2582	5514885	33185,20	7031	14,67	14,24
Média		24703,08	1069	2358537	21519,84	1961	9,78	8,52
Média global		24574,33	1208	2274276	21443,87	1797	9,76	8,14

FONTE: A autora (2019)

Tabela 3.15: RESULTADOS MÉDIOS PARA INSTÂNCIAS COM 10 CLIENTES E 3 DEPÓSITOS: COMPARAÇÃO ENTRE O PLR-DT E O PRV-DT SEQUENCIAL

Tipo	Instância	PLR-DT		PRV-DT						
		Tempo(s)	Solução	Depósito 1		Depósito 2		Depósito 3		Tempo total(s)
				Tempo(s)	Solução	Tempo(s)	Solução	Tempo(s)	Solução	
Pequena	1	29,78	7193	35,72	8165	6,74	7193	5,36	7350	47,82
	2	6,85	7740	7,06	9237	7,93	8398	3,13	7740	18,12
	3	2,01	6319	3,73	7856	9,45	8859	1,24	6319	14,42
	4	174,43	7120	78,15	7120	737,52	8727	133,73	7516	949,40
	5	40,06	6780	15,64	6780	7204,73	11039	812,64	8593	8033,01
Média	1	51,48	7407	39,07	8096	319,70	9229	14,83	7407	373,60
	2	901,25	8390	2716,67	9549	864,86	8805	317,93	8390	3899,46
	3	7,14	6236	1035,03	7874	3313,91	9066	4,12	6236	4353,06
	4	120,41	7021	119,96	7021	16,59	8388	31,81	7181	168,36
	5	48,29	7209	26,37	7209	517,00	11030	86,69	9004	630,06
Grande	1	74,28	7243	65,75	8028	867,94	9275	19,61	7243	953,30
	2	267,49	8326	123,82	9558	200,54	8779	107,76	8326	432,12
	3	6,43	6151	8,84	7666	10,48	8482	5,85	6151	25,17
	4	308,09	6963	191,67	6963	1142,61	8270	86,06	6973	1420,34
	5	72,94	7082	21,83	7082	2184,93	11218	298,74	9042	2505,50

FONTE: A autora (2019)

com 10 clientes e 3 depósitos. Esse tempo corresponde a apenas cerca de 15% do tempo necessário na abordagem sequencial.

Tabela 3.16: RESULTADOS MÉDIOS PARA INSTÂNCIAS COM 10 CLIENTES E 5 DEPÓSITOS: COMPARAÇÃO ENTRE O PLR-DT E O PRV-DT SEQUENCIAL

Tipo	Instância	PLR-DT		PRV-DT										
		Tempo(s)	Solução	Depósito 1		Depósito 2		Depósito 3		Depósito 4		Depósito 5		Tempo total(s)
				Tempo(s)	Solução	Tempo(s)	Solução	Tempo(s)	Solução	Tempo(s)	Solução	Tempo(s)	Solução	
Pequena	1	379,55	6723	26,93	6847	1760,60	9427	9,90	6723	40,67	6788	108,13	7541	1946,23
	2	98,99	5595	98,59	6240	16,89	5689	6,86	5711	38,89	5595	1271,05	8065	1432,28
	3	493,62	7519	260,16	7809	52,20	7519	764,99	8816	1120,48	9057	248,07	8138	2445,90
	4	285,55	9145	78,38	9145	140,43	9553	56,64	10848	98,06	10467	1,02	10780	374,53
	5	509,65	9038	287,67	9796	66,72	9618	60,04	11042	17,10	9038	33,84	9252	465,37
Média	1	26,59	6532	3,18	6532	52,08	8378	11,81	6678	2,35	6589	7,83	6827	77,25
	2	141,61	5537	16,44	6369	5,94	5537	49,82	5912	36,79	5742	49,12	8717	158,11
	3	232,15	7907	1940,63	8989	34,63	7907	86,15	9322	95,21	9535	44,25	8531	2200,87
	4	11361,40	10517	2351,43	10517	3168,67	11030	226,76	11042	262,43	10537	31,28	11389	6040,57
	5	136,63	8449	52,09	8449	13,29	8474	3043,11	10514	266,78	8983	16,03	8458	3391,30
Grande	1	118,52	6119	16,72	6282	52,69	6876	13,66	6119	60,54	6555	13,34	6350	156,95
	2	1635,56	5798	948,40	6529	227,05	5798	308,82	5890	3554,86	6434	36577,30	8626	41616,43
	3	209,73	7568	75,84	7568	50,75	7654	16,64	8570	12,73	8801	22,20	8154	178,16
	4	580,39	8409	223,59	8409	188,94	8653	17286,60	10791	10380,50	10234	21650,10	11128	49729,73
	5	482,36	8883	107,93	9061	51,30	9117	26,18	10530	92,28	9053	30,19	8883	307,88

FONTE: A autora (2019)

Os resultados da Tabela 3.16 também mostram que o PLR-DT supera claramente o PRV-DT seqüencial. A abordagem do PLR-DT apresenta melhores resultados em 73,33% das instâncias com 10 clientes e 5 potenciais depósitos. Os dados mostram que instâncias com um número pequeno de intervalos de tempo obtêm a solução ótima cerca de 96% mais rápido, quando comparado com o tempo exigido pela abordagem seqüencial do PRV-DT. Mesmo em instâncias com um grande número de intervalos, esse tempo ainda é 73% mais rápido.

### 3.5 CONCLUSÕES

Este trabalho apresenta a primeira formulação matemática para o problema de localização e roteamento dependente do tempo, contribuindo com a literatura da área de logística integrada. Para obter soluções de qualidade, em um tempo computacional aceitável, e fornecer um conjunto de soluções iniciais, propôs-se também um algoritmo metaheurístico. O desempenho desse algoritmo foi comparado com a formulação exata, em um conjunto de instâncias baseadas em dados reais. Mostrou-se que, para problemas complexos como esse, é importante integrar métodos heurísticos e formulações exatas. As heurísticas construtivas propostas foram capazes de gerar rotas diversificadas e soluções iniciais que podem ser combinadas e otimizadas através da resolução de um problema de particionamento de conjuntos. Além disso, a formulação matemática apresentada é capaz de fornecer soluções de qualidade para instâncias com até 100 clientes e 15 intervalos de tempo, onde as condições de tráfego mudam de hora em hora. A inclusão de um conjunto de desigualdades válidas fornece melhores limites inferiores, enquanto um conjunto de soluções iniciais permite melhorias adicionais.

## 4 PROBLEMA DE LOCALIZAÇÃO E ROTEAMENTO DEPENDENTE DO TEMPO COM DIMENSIONAMENTO DE FROTA HETEROGÊNEA

Neste capítulo, apresenta-se o problema de localização e roteamento dependente do tempo com dimensionamento de frota heterogênea. Ele incorpora aspectos como limitação de capacidade nos depósitos, frota heterogênea e não limitada e flexibilidade no horário de saída dos veículos. Na Seção 4.1, define-se formalmente o problema e propõe-se um modelo matemático de programação linear inteira mista para sua resolução. Na Seção 4.2, expõe-se também uma abordagem de resolução meta-heurística e explicam-se os detalhes da sua estrutura. Na Seção 4.3, especificam-se os procedimentos realizados na execução dos testes computacionais e na geração das instâncias. Na Seção 4.4, apresentam-se e discutem-se os resultados alcançados. Por fim, na Seção 4.5, encerra-se com as conclusões.

### 4.1 DESCRIÇÃO DO PROBLEMA E FORMULAÇÃO MATEMÁTICA

Define-se o PLRDFH-DT a partir de um grafo direcionado  $G(V, A)$ , onde  $V$  representa o conjunto de vértices e  $A$  o conjunto de arcos. Sejam  $V_d$  o conjunto de todos os potenciais depósitos,  $V_c$  o conjunto de todos os clientes e  $V_t$  um conjunto de vértices fictícios, denominados terminais, tal que  $V_d \cap V_c = \emptyset$ ,  $V_d \cap V_t = \emptyset$ ,  $V_c \cap V_t = \emptyset$  e  $V = V_d \cup V_c \cup V_t$ . Define-se  $K = \{1, 2, \dots, k, \dots\}$  como o conjunto dos  $|K|$  diferentes tipos de veículo da frota, previamente conhecidos, com capacidade individual limitada por  $Q_k$  unidades. Define-se ainda  $\delta_k(i)$  como um subconjunto unitário do terminal que está vinculado ao veículo do tipo  $k \in K$  e ao depósito  $i \in V_d$ , ou seja,  $\delta_k(i) \subseteq V_t$ . Assim, têm-se um terminal  $v \in \delta_k(i)$ ,  $i \in V_d$  associado a cada tipo de veículo da frota, em cada potencial depósito.

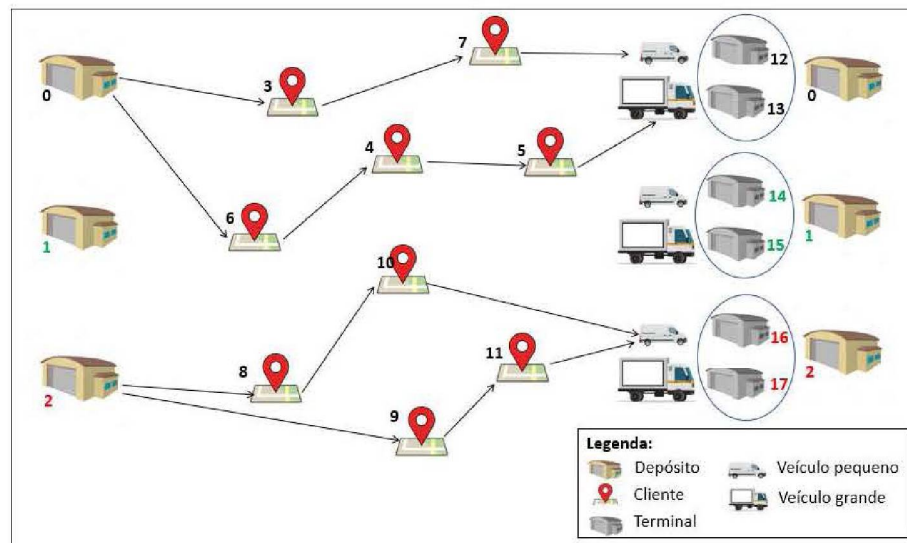
Sejam ainda  $A_{dc}$ ,  $A_{cc}$ ,  $A_{ct}$  conjuntos de arcos, tais que cada arco  $(i, j)$  é dado a partir de um dos produtos cartesianos  $A_{dc} = V_d \times V_c$ ;  $A_{cc} = V_c \times V_c$ ,  $i \neq j$ , e  $A_{ct} = \bigcup_{k \in K} \bigcup_{i \in V_d} \{V_c \times \delta_k(i)\}$ , de forma que  $A = A_{dc} \cup A_{cc} \cup A_{ct}$ . Seja ainda  $H$  o conjunto de intervalos de tempo. Define-se o intervalo como um período de tempo para o qual o padrão de tráfego é constante. A cada arco  $(i, j) \in A$  está associado um tempo de deslocamento  $t_{ij}^h$ , que depende do intervalo  $h \in H$ . O horizonte de planejamento é de um dia, ou uma fração do dia, dividido em intervalos de tempo com duração igual a  $\bar{T}$  unidades. Para  $h \in H = \{0, 1, \dots, h, \dots, m\}$ ,  $m + 1$  representa o número de intervalos diários. Dessa forma,  $[h\bar{T}, (h + 1)\bar{T} - \varepsilon]$  representa o intervalo associado a  $h$ , onde  $\varepsilon$  indica uma unidade de tempo.

A demanda de cada cliente  $i \in V_c$  é denotada por  $q_i$  e o seu tempo de serviço por  $s_i$ . O custo fixo de utilização de cada veículo do tipo  $k \in K$  é representado por  $F_k$ ;  $W_i$  denota a capacidade do depósito  $i \in V_d$  e  $O_i$  representa seu custo de abertura. Por fim,  $C$  denota o custo por unidade de tempo  $t_{ij}^h$ , nos arcos  $(i, j) \in A$ , com  $h \in H$ , independente do intervalo de tempo ou tipo de veículo usado.

Sabendo que todos os clientes precisam ser atendidos, o objetivo é minimizar o custo total, que é composto por três parcelas: custo de abertura dos depósitos, custo fixo de utilização dos veículos e custo de execução das rotas. Ao final, identificam-se o(s) depósito(s) com a melhor localização, as rotas que devem ser executadas para atendimento dos clientes e a frota que será utilizada (número de veículos e tipo).

Para ilustrar a função dos terminais, considere como exemplo um problema com três potenciais depósitos, nove clientes e uma frota heterogênea composta por dois tipos de veículo (pequeno e grande). Uma solução para esse problema é representada através da Figura 4.1.

FIGURA 4.1 - REPRESENTAÇÃO DA SOLUÇÃO DE UM PLRDFH-DT



FONTE: A autora (2019)

Note que a cada tipo de veículo da frota há um terminal associado para cada potencial depósito. Ele serve para indicar a utilização de um tipo específico de veículo e seu retorno ao depósito de origem. No exemplo em questão, um veículo pequeno e um grande são usados para atender os clientes designados ao primeiro depósito, enquanto dois veículos pequenos são usados para executar as rotas dos clientes designados ao segundo depósito. O retorno de cada um se dá a partir do respectivo terminal, que está associado com o tipo de veículo usado e com o depósito de origem. Isso possibilita, sobretudo, controlar a capacidade de cada tipo de veículo disponibilizado na frota. Além disso, facilita a identificação do número de veículos (por tipo) que foram utilizados, o tempo de execução e a demanda total dos clientes atendidos em cada rota.

Para o modelo matemático, definem-se as seguintes variáveis:

- a variável binária  $x_{ij}^h$  assume valor 1 se o arco  $(i, j)$  é atravessado por um veículo no intervalo de tempo  $h$ , e 0 em caso contrário;
- a variável binária  $z_{ij}$  assume valor 1 se o arco  $(i, j)$  é atravessado por um veículo, e 0 em caso contrário;

- a variável binária  $w_i$  assume valor 1 se o potencial depósito  $i$  é aberto, e 0 em caso contrário;
- a variável binária  $y_i^h$  assume valor 1 se uma rota parte do cliente  $i \in V_c$  no intervalo de tempo  $h \in H$ , e 0 em caso contrário;
- a variável contínua  $a_i$  indica o momento de partida no cliente  $i \in V_c$ ;
- a variável contínua  $u_i$  indica a demanda acumulada entregue até a saída do veículo no cliente  $i \in V_c$ , caso esta seja igual a capacidade do veículo. Se a demanda total for inferior, essa variável pode assumir uma folga, que está associada à capacidade não utilizada;
- a variável binária  $g_{ij}^h$  assume valor 1 se uma rota começa no arco  $(i, j)$ , com  $i \in V_d$ ,  $j \in V_c$ , e no intervalo de tempo  $h \in H$ , e 0 em caso contrário;
- a variável contínua  $b_{ij}$  indica o momento de partida no depósito  $i \in V_d$  ou de chegada no terminal  $j \in V_t$ , para  $(i, j) \in A \setminus A_{cc}$ , caso o arco  $(i, j)$  seja atravessado por algum veículo;
- a variável contínua  $l_{ij}$  indica a demanda total entregue pelo veículo na chegada ao terminal  $j \in V_t$ , caso o arco  $(i, j)$  seja atravessado. Assim como na variável anterior, esta também pode assumir uma folga;
- a variável contínua  $f_i$  assume o valor do índice do depósito ao qual o vértice  $i \in V$  está vinculado na roteirização.

Note que, em relação ao modelo matemático da Seção 3.1, foram adicionadas novas variáveis. Aqui, a variável  $g_{ij}^h$  substitui a variável  $y_i^h$  nos arcos que iniciam nos depósitos e finalizam nos clientes. Da mesma forma, a variável  $b_{ij}$  substitui a variável  $a_i$  nesses mesmos arcos e também nos arcos que partem dos clientes em direção aos terminais. Com a incorporação da flexibilidade no horário de saída dos veículos, estes podem iniciar a rota a partir de um mesmo depósito, mas em momentos e, consequentemente, intervalos distintos. Dessa forma,  $g_{ij}^h$  controla o intervalo de saída de cada veículo no depósito e  $b_{ij}$  controla o momento de saída no depósito e de chegada nos terminais. A variável  $l_{ij}$  substitui a variável  $u_i$  nos arcos que ligam clientes aos terminais, como forma de controlar a capacidade utilizada em cada veículo, uma vez que o problema considera frota heterogênea. Por fim, a variável  $f_i$  foi incorporada para garantir que os veículos retornem ao depósito de origem, uma vez que o modelo possibilita a abertura de mais de um depósito.

A Tabela 4.1 resume a notação usada na formulação matemática do problema.

Tabela 4.1: NOTAÇÃO USADA NO PLRDFH-DT

Parâmetros	
$m + 1$	Número de intervalos de tempo do horizonte de planejamento
$\bar{T}$	Duração de cada intervalo de tempo $h \in H$
$\varepsilon$	Uma unidade de tempo considerada
$q_i$	Demanda de cada cliente $i \in V_c$
$s_i$	Tempo de serviço no cliente $i \in V_c$
$Q_k$	Capacidade de cada veículo do tipo $k \in K$
$F_k$	Custo fixo de utilização de cada veículo do tipo $k \in K$
$W_i$	Capacidade do depósito $i \in V_d$
$O_i$	Custo de abertura do depósito $i \in V_d$
$C$	Custo por unidade de tempo de deslocamento $t_{ij}^h$ no arco $(i, j) \in A$ , e em $h \in H$
$t_{ij}^h$	Tempo de viagem no arco $(i, j) \in A$ , no intervalo de tempo $h \in H$
Conjuntos	
$V_c$	Conjunto de clientes
$V_d$	Conjunto de potenciais depósitos
$V_t$	Conjunto de terminais
$K$	Conjunto de tamanhos de veículos da frota
$H$	Conjunto de intervalos de tempo
$A$	Conjunto de arcos
$\delta_k(i)$	Subconjunto de um único terminal para cada tipo de veículo $k \in K$ e cada potencial depósito $i \in V_d$
Variáveis de decisão	
$x_{ij}^h$	Se um veículo atravessa, ou não, o arco $(i, j) \in A$ no intervalo de tempo $h \in H$
$z_{ij}$	Se um veículo atravessa, ou não, o arco $(i, j) \in A$
$w_i$	Se o depósito $i \in V_d$ é aberto, ou não
$y_i^h$	Se algum veículo parte, ou não, do cliente $i \in V_c$ no intervalo de tempo $h \in H$
$s_{ij}^h$	Se alguma rota começa, ou não, no arco $(i, j), i \in V_d, j \in V_c$ , no intervalo de tempo $h \in H$
$a_i$	Momento de partida no cliente $i \in V_c$
$b_{ij}$	Momento de partida no depósito $i \in V_d$ ou de chegada no terminal $j \in V_t$ para $(i, j) \in A \setminus A_{cc}$
$u_i$	Demanda acumulada entregue na saída do veículo no cliente $i \in V_c$ , podendo ser acrescida de uma folga
$l_{ij}$	Demanda total entregue na chegada do veículo ao terminal $j \in V_t$ , podendo ser acrescida de uma folga
$f_i$	Índice do depósito ao qual o vértice $i \in V$ está vinculado na roteirização
Índices	
$h$	Intervalo de tempo
$i, j, v$	Vértices
$k$	Tipo de veículo

FONTE: A autora (2019)

A formulação matemática proposta é descrita na sequência:

$$\min \sum_{(i,j) \in A} \sum_{h \in H} C t_{ij}^h x_{ij}^h + \sum_{i \in V_d} O_i w_i + \sum_{k \in K} \sum_{j \in V_c} \sum_{v \in \delta_k(i)} F_k z_{jv} \quad (4.1)$$

Sujeito a:

$$\sum_{i \in (V_c \setminus \{j\}) \cup V_d} z_{ij} = 1, \quad \forall j \in V_c \quad (4.2)$$

$$\sum_{j \in (V_c \setminus \{i\}) \cup V_t} z_{ij} = 1, \quad \forall i \in V_c \quad (4.3)$$

$$\sum_{j \in V_c} z_{ij} \leq |V_c| w_i, \quad \forall i \in V_d \quad (4.4)$$

$$\sum_{j \in V_c} z_{ij} = \sum_{j \in V_c} \sum_{k \in K} \sum_{v \in \delta_k(i)} z_{jv}, \quad \forall i \in V_d \quad (4.5)$$

$$u_i - u_j + \max_{k \in K} \{Q_k\} z_{ij} \leq \max_{k \in K} \{Q_k\} - q_j, \quad \forall i, j \in V_c, i \neq j \quad (4.6)$$

$$q_i \leq u_i \leq \max_{k \in K} \{Q_k\}, \quad \forall i \in V_c \quad (4.7)$$

$$l_{jv} \leq u_j, \quad \forall j \in V_c, \forall v \in V_t \quad (4.8)$$

$$l_{jv} \geq u_j - \max_{k \in K} \{Q_k\} (1 - z_{jv}), \quad \forall j \in V_c, \forall v \in V_t \quad (4.9)$$

$$l_{ij} \leq Q_k z_{ij}, \quad \forall i \in V_c, \forall j \in \delta_k(v), \forall k \in K, \forall v \in V_d \quad (4.10)$$

$$\sum_{j \in V_c} \sum_{\substack{v \in \delta_k(i) \\ k \in K}} l_{jv} \leq W_i, \quad \forall i \in V_d \quad (4.11)$$

$$\sum_{h \in H} x_{ij}^h = z_{ij}, \quad \forall (i, j) \in A \quad (4.12)$$

$$x_{ij}^h \leq y_i^h, \quad \forall (i, j) \in A \setminus A_{dc}, \forall h \in H \quad (4.13)$$

$$x_{ij}^h \leq g_{ij}^h, \quad \forall (i, j) \in A_{dc}, \forall h \in H \quad (4.14)$$

$$\sum_{h \in H} g_{ij}^h = z_{ij}, \quad \forall (i, j) \in A_{dc} \quad (4.15)$$

$$\sum_{h \in H} y_i^h = 1, \quad \forall i \in V_c \quad (4.16)$$

$$a_j \geq b_{ij} + s_j + t_{ij}^h - 2\bar{T}|H|(1 - x_{ij}^h), \quad \forall i \in V_d, \forall j \in V_c, \forall h \in H \quad (4.17)$$

$$a_j \leq b_{ij} + s_j + t_{ij}^h + \bar{T}|H|(1 - x_{ij}^h), \quad \forall i \in V_d, \forall j \in V_c, \forall h \in H \quad (4.18)$$

$$a_j \geq a_i + s_j + t_{ij}^h - 2\bar{T}|H|(1 - x_{ij}^h), \quad \forall i, j \in V_c, i \neq j, \forall h \in H \quad (4.19)$$

$$a_j \leq a_i + s_j + t_{ij}^h + \bar{T}|H|(1 - x_{ij}^h), \quad \forall i, j \in V_c, i \neq j, \forall h \in H \quad (4.20)$$



$$b_{jv} \geq a_j + t_{jv}^h - 2\bar{T}|H|(1 - x_{jv}^h), \quad \forall j \in V_c, \forall v \in V_t, \forall h \in H \quad (4.21)$$

$$b_{jv} \leq a_j + t_{jv}^h + \bar{T}|H|(1 - x_{jv}^h), \quad \forall j \in V_c, \forall v \in V_t, \forall h \in H \quad (4.22)$$

$$b_{ij} \leq (\bar{T}|H| - \varepsilon)z_{ij}, \quad \forall (i, j) \in A \setminus A_{cc} \quad (4.23)$$

$$\sum_{h \in H} h\bar{T}y_i^h \leq a_i \leq \sum_{h \in H} h\bar{T}y_i^h + \bar{T} - \varepsilon, \quad i \in V_c \quad (4.24)$$

$$\sum_{h \in H} h\bar{T}g_{ij}^h \leq b_{ij} \leq \sum_{h \in H} h\bar{T}g_{ij}^h + (\bar{T} - \varepsilon), \quad \forall (i, j) \in A_{dc} \quad (4.25)$$

$$f_i = i, \quad \forall i \in V_d \quad (4.26)$$

$$f_j = f_i, \quad \forall j \in \delta_k(i), \forall k \in K, \forall i \in V_d \quad (4.27)$$

$$f_j \geq f_i + |V_d|(z_{ij} - 1), \quad \forall (i, j) \in A \quad (4.28)$$

$$f_j \leq f_i + |V_d|(1 - z_{ij}), \quad \forall (i, j) \in A \quad (4.29)$$

$$w_i \in \{0, 1\}, \quad \forall i \in V_d \quad (4.30)$$

$$z_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A \quad (4.31)$$

$$y_i^h \in \{0, 1\}, \quad \forall i \in V_c, \forall h \in H \quad (4.32)$$

$$g_{ij}^h \in \{0, 1\}, \quad \forall (i, j) \in A_{dc}, \forall h \in H \quad (4.33)$$

$$x_{ij}^h \in \{0, 1\}, \quad \forall (i, j) \in A, \forall h \in H \quad (4.34)$$

$$a_i \in \mathbb{R}_+, \quad \forall i \in V_c \quad (4.35)$$

$$b_{ij} \in \mathbb{R}_+, \quad \forall (i, j) \in A \setminus A_{cc} \quad (4.36)$$

$$u_i \in \mathbb{R}_+, \quad \forall i \in V_c \quad (4.37)$$

$$l_{ij} \in \mathbb{R}_+, \quad \forall (i, j) \in A_{ct} \quad (4.38)$$

$$f_i \in \mathbb{R}_+, \quad \forall i \in V. \quad (4.39)$$

O problema consiste em minimizar a função objetivo, dada pela soma total dos custos de execução das rotas (dependente do tempo de viagem), dos custos fixos de abertura dos depósitos e de utilização dos veículos, conforme expresso em (4.1). As restrições (4.2) e (4.3) são restrições de designação, que garantem que cada cliente seja visitado exatamente uma vez. O conjunto de restrições (4.4) impõem que as rotas sejam executadas apenas a partir dos depósitos abertos, e o conjunto (4.5) equilibra o fluxo de veículos entre o depósito e os terminais a ele associados. Note que há um terminal associado a cada tipo de veículo da frota, para cada potencial depósito. Cada rota criada é finalizada em um dos terminais, de acordo com a demanda por ela acumulada. As restrições (4.6) e (4.7) são uma extensão da eliminação de sub-rotas de Miller-Tucker-Zemlin, propostas por Kulkarni e Bhave (1985). Através das restrições (4.8) e (4.9) é possível controlar a demanda total entregue pelo veículo, enquanto as restrições (4.10) restringem a capacidade de cada tipo de veículo da frota. Restrições do conjunto (4.11) garantem que a demanda dos clientes vinculados ao depósito  $i \in V_d$  não ultrapasse a sua capacidade.

As restrições (4.12) impõem que um único intervalo de tempo seja considerado na travessia de cada arco  $(i, j)$ . Se um arco  $(i, j)$  é atravessado por um veículo no intervalo  $h$ , então  $h$  é o intervalo de tempo considerado na saída do veículo do vértice  $i$ , de acordo com a imposição das restrições (4.13) e (4.14). Apenas um intervalo de tempo  $h$  é usado na saída do veículo do depósito ou cliente  $i$ , conforme garantem as restrições (4.15) e (4.16), respectivamente. Restrições dos conjuntos (4.17) e (4.18) controlam o momento de saída dos veículos no primeiro cliente de cada rota. O momento de saída no vértice  $j$ , após a travessia do arco  $(i, j)$ , inclui o momento de saída no depósito  $i$ , o tempo de deslocamento necessário para atravessar o arco  $(i, j)$ , e o tempo de serviço no cliente  $j$ . Note que nas restrições do conjunto (4.17) é necessário multiplicar o último termo por 2 para garantir que ele ainda atue como um *Big-M* válido, caso  $b_{ij} + s_j + t_{ij}^h$  seja maior que o horizonte de planejamento. Da mesma forma, as restrições dos conjuntos (4.19) e (4.20) executam esse controle nos demais clientes, enquanto as restrições (4.21) e (4.22) são usadas para controlar o momento de chegada no terminal  $v$ . Restrições do conjunto (4.23) asseguram que os veículos executem suas rotas dentro do horizonte de planejamento, enquanto as restrições (4.24) e (4.25) limitam o momento de saída nos vértices ao intervalo de tempo correspondente. Os conjuntos de restrições (4.26) – (4.29) garantem que cada veículo retorne ao depósito de origem. Finalmente, as restrições (4.30) – (4.39) forçam as condições de integralidade e de não-negatividade das variáveis.

Note que as restrições (4.13) e (4.32), (4.14) e (4.33) implicam  $x_{ij}^h \leq 1$ . Mais ainda, conforme veremos no teorema abaixo, a não-negatividade das variáveis  $x_{ij}^h$  já garante a integralidade, ou seja, assumindo não-negatividade das variáveis  $x_{ij}^h$ , as restrições (4.34) são redundantes.

**Teorema 4.1** *Suponha que as restrições (4.12) – (4.16) e (4.30) – (4.33) sejam satisfeitas. Se*

$$x_{ij}^h \geq 0, \quad \forall (i, j) \in A, \forall h \in H, \quad (4.40)$$

*então  $x_{ij}^h \in \{0, 1\}, \forall (i, j) \in A, \forall h \in H$ .*

*Demonstração:* Considere  $(i, j) \in A$  e  $h^* \in H$  arbitrários. Se  $g_{ij}^{h^*} = 0$ , onde  $(i, j) \in A_{dc}$ , então  $x_{ij}^{h^*} = 0$  por (4.14) e (4.40). Caso contrário, por (4.33), tem-se que  $g_{ij}^{h^*} = 1$ . Assim, de (4.15), (4.31) e (4.33), segue que  $g_{ij}^h = 0$ , e portanto  $x_{ij}^h = 0, \forall h \neq h^*$ . Da mesma forma, se  $y_i^{h^*} = 0$ , onde  $i \in V_c$  então  $x_{ij}^{h^*} = 0 \forall (i, j) \in A \setminus A_{dc}$  por (4.13) e (4.40). Caso contrário, por (4.32), tem-se que  $y_i^{h^*} = 1$ . Assim, de (4.16) e (4.32), segue que  $y_i^h = 0$ , e portanto  $x_{ij}^h = 0, \forall h \neq h^*$ . Logo, usando (4.12) e (4.31), obtém-se  $x_{ij}^{h^*} = \sum_{h \in H} x_{ij}^h = z_{ij} \in \{0, 1\}$ , provando a afirmação do teorema. Portanto,  $x_{ij}^h \in \{0, 1\}, \forall (i, j) \in A, \forall h \in H$ .

#### 4.1.1 Desigualdades válidas

Com o intuito de fortalecer o modelo, define-se também um conjunto de desigualdades válidas, conforme segue.

Como feito em Kara, Laporte e Bektaş (2004), as restrições (4.6) podem ser ajustadas, resultando em (4.41):

$$u_i - u_j + \max_{k \in K} \{Q_k\} z_{ij} + (\max_{k \in K} \{Q_k\} - q_i - q_j) z_{ji} \leq \max_{k \in K} \{Q_k\} - q_j, \quad \forall i, j \in V_c, i \neq j. \quad (4.41)$$

O tamanho do problema pode ser reduzido eliminando algumas variáveis que estão associadas com o intervalo de saída dos veículos nos vértices. Isso acontece quando o tempo de serviço no cliente  $i$  mais o menor tempo para atravessar o arco entre algum depósito (ou outro cliente) e o cliente  $i$  é superior a duração do intervalo. Também ocorre quando o tempo de serviço em um cliente mais o menor tempo para atravessar um arco entre algum depósito e algum cliente é maior que a duração do intervalo. Essas situações são consideradas pelas restrições (4.42) e (4.43), para  $h = 0$ .

$$x_{ij}^0 = 0 \quad \forall i \in V_c \mid \left( \min_{a \in (V_c \setminus \{i\}) \cup V_d} \{t_{ai}^0\} + s_i \right) \geq \bar{T}, \quad \forall j \in V_c \cup V_t, j \neq i \quad (4.42)$$

$$x_{ij}^0 = 0 \quad \forall i \in V_c \mid s_i \geq \bar{T} - \min_{\substack{a \in V_d \\ b \in V_c}} \{t_{ab}^0\}, \quad \forall j \in V_c \cup V_t, j \neq i. \quad (4.43)$$

Essa ponderação serve para qualquer intervalo de tempo  $h \in H$ , conforme generalizado em (4.44) e (4.45):

$$x_{ij}^h = 0 \quad \forall i \in V_c \mid \left( \min_{a \in (V_c \setminus \{i\}) \cup V_d} \{t_{ai}^h\} + s_i \right) \geq (h+1)\bar{T}, \quad \forall j \in V_c \cup V_t, j \neq i, \forall h \in H \quad (4.44)$$

$$x_{ij}^h = 0 \quad \forall i \in V_c \mid s_i \geq (h+1)\bar{T} - \min_{\substack{a \in V_d \\ b \in V_c}} \{t_{ab}^h\}, \quad \forall j \in V_c \cup V_t, j \neq i, \forall h \in H. \quad (4.45)$$

Da mesma forma, para o último intervalo do horizonte de planejamento também pode ser possível remover variáveis associadas com os terminais. Isso ocorre quando o tempo para atravessar um arco, no último intervalo, e chegar ao terminal é maior que a duração do intervalo. Ou ainda, se o menor tempo para chegar ao cliente, somado com seu tempo de serviço e com o menor tempo para ir desse cliente ao terminal, é maior que a duração do intervalo.

$$x_{ij}^m = 0, \quad \forall i \in V_c, \forall j \in V_t \mid t_{ij}^m \geq \bar{T} \quad (4.46)$$

$$x_{ij}^m = 0, \quad \forall j \in V_c \mid \left( \min_{a \in V_c \setminus \{j\}} \{t_{aj}^m\} + s_j + \min_{b \in V_t} \{t_{jb}^m\} \right) \geq \bar{T}, \quad \forall i \in V_c, i \neq j. \quad (4.47)$$

Questões do modelo associadas à roteirização podem ser melhoradas evitando que sub-rotas com dois ou três clientes sejam criadas:

$$z_{ij} + z_{ji} \leq 1, \quad \forall i, j \in V_c, i \neq j \quad (4.48)$$

$$z_{ij} + z_{ji} + z_{iv} + z_{vi} + z_{jv} + z_{vj} \leq 2, \quad \forall i, j, v \in V_c, i \neq j, i \neq v, j \neq v. \quad (4.49)$$

Além disso, limites inferiores associados a cada uma das parcelas de custo da função objetivo podem ser estabelecidos. As restrições (4.50) estabelecem um limite inferior para os custos de abertura dos depósitos. Seja  $o'$  o custo mínimo necessário para abertura de depósitos, obtido como solução de um problema de empacotamento, onde consideram-se a demanda dos clientes e a capacidade dos potenciais depósitos.

$$\sum_{i \in V_d} O_i w_i \geq o'. \quad (4.50)$$

O outro limite inferior que pode ser definido trata dos custos fixos relacionados à utilização da frota. Seja  $f'$  o custo mínimo necessário para utilização dos veículos, considerando

a capacidade dos diferentes tipos de veículo disponíveis e a demanda dos clientes, obtido como solução de um problema de empacotamento.

$$\sum_{k \in K} \sum_{j \in V_c} \sum_{\substack{v \in \delta_k^+(i) \\ i \in V_d}} F_k z_{jv} \geq f'. \quad (4.51)$$

Ainda, pode-se estabelecer um limite inferior para o custo de execução das rotas, a partir de uma soma de menores tempos. Seja  $k'$  o número mínimo de veículos com capacidade  $\max_{k \in K} \{Q_k\}$ , necessários para que a demanda dos clientes seja atendida, sendo  $k'$  a solução de um problema de empacotamento. Seja  $s_{di} = \min_{h \in H} \{t_{di}^h\}$ ,  $\forall d \in V_d, \forall i \in V_c$ . Para cada  $d \in V_d$ , seja  $s_{di_n} \mid s_{di_1} \leq s_{di_2} \leq \dots \leq s_{di_{|V_c|}}$ . Similarmente, seja  $r_{id} = \min_{h \in H} \{t_{id}^h\}$ , com  $r_{id_n} \mid r_{id_1} \leq r_{id_2} \leq \dots \leq r_{id_{|V_c|}}$   $\forall d \in V_d$ . Agora,  $\forall d \in V_d$ , seja ainda  $f_{d_n} = s_{di_n} + r_{id_n}$ , com  $f_{d_n} \mid f_{d_1} \leq f_{d_2} \leq \dots \leq f_{d_{|V_c|}}$ . Finalmente, seja  $g_c = \min_{\substack{a \in V_c \setminus \{i\} \\ h \in H}} \{t_{ca}^h\}$ ,  $\forall c \in V_c$  e  $g_{c_n} \mid g_{c_1} \leq g_{c_2} \leq \dots \leq g_{c_{|V_c|}}$ .

$$\sum_{(i,j) \in A} \sum_{h \in H} C t_{ij}^h x_{ij}^h \geq \sum_{n=1}^{k'} C f_{d_n} + \sum_{n=1}^{|V_c|-k'} C g_{c_n}. \quad (4.52)$$

Desigualdades válidas relacionadas aos fluxos de saída dos veículos dos depósitos também podem ser criadas. As restrições (4.53) e (4.54) estabelecem que o fluxo mínimo de veículos saindo dos depósitos equivale a quantidade mínima de veículos de maior capacidade, necessários para atender os clientes. Novamente, seja  $k'$  o número mínimo de veículos com capacidade  $\max_{k \in K} \{Q_k\}$ , necessários para que a demanda dos clientes seja atendida, onde  $k'$  é resultado de um problema de empacotamento.

$$\sum_{i \in V_d} \sum_{j \in V_c} \sum_{h \in H} x_{ij}^h \geq k' \quad (4.53)$$

$$\sum_{i \in V_d} \sum_{j \in V_c} z_{ij} \geq k'. \quad (4.54)$$

Por fim, a restrição (4.55) estabelece que, nos depósitos abertos, o fluxo de saída de veículos deve ser, no mínimo, igual a um.

$$\sum_{i \in V_d} \sum_{j \in V_c} \sum_{h \in H} g_{ij}^h \geq w_l. \quad (4.55)$$

## 4.2 ALGORITMO META-HEURÍSTICO

A estrutura geral do algoritmo meta-heurístico é baseada em busca evolucionária. Ele foi inspirado nas ideias propostas por Koç et al. (2016), porém considera o aspecto da dependência de tempo. O algoritmo engloba três fases principais, assim denominadas: inicialização, evolução

e intensificação. A primeira trata da geração de uma população inicial. A segunda envolve a criação de novas gerações de soluções e a seleção dos indivíduos. A fase final compreende um processo de melhoria adicional, aplicado às soluções da elite.

O Algoritmo 8 descreve as principais etapas de cada fase da meta-heurística. A inicialização consiste na criação de soluções iniciais, a partir da designação de clientes aos depósitos e posterior construção de rotas (linha 2). Em seguida, essa população inicial é incrementada, até atingir  $\eta$  indivíduos, através da aplicação de operadores de remoção e inserção (linha 4). Na fase de evolução, um subconjunto de indivíduos é selecionado da população atual para atuar como pais na geração seguinte (linha 8). Destes, dois são escolhidos para cada geração por cruzamento (linha 9). Na mutação direcionada, seleciona-se um único pai, e conduz-se a criação de um novo indivíduo utilizando operadores de remoção e inserção (linha 10). Na sequência, executa-se um procedimento de seleção da população sobrevivente (linha 11). Quando o número  $\delta$  de iterações sem melhoria na população corrente é atingido, encerra-se esta fase (linha 12). Inicia-se então a intensificação. Operadores de troca baseados em demanda e tempo de serviço são aplicados a todo conjunto de soluções correntes (linha 15). Depois, seleciona-se a população remanescente (linha 16). Quando não for mais possível gerar melhoria em algum desses indivíduos, finaliza-se essa etapa (linha 17). Sobre as  $\theta$  soluções da elite (linha 18) é aplicada uma heurística de melhoria em relação ao momento de saída de cada veículo e a sequência de clientes atendidos na rota (linha 20). Após a execução desse procedimento, o custo da solução é atualizado (linha 21). Por fim, retorna-se a melhor solução encontrada (linha 22) e encerra-se o algoritmo meta-heurístico.

Cada uma dessas fases é descrita com mais detalhes na sequência.

#### 4.2.1 Inicialização

O procedimento de inicialização do algoritmo meta-heurístico é baseado na geração de soluções iniciais e, a partir delas, na criação de outras soluções que irão compor a população inicial, conforme especificado a seguir.

A etapa de geração de soluções iniciais compreende a designação dos clientes aos depósitos, seguida pela construção de rotas. A designação é baseada no tempo de deslocamento entre depósito e cliente. Como esse tempo varia ao longo do horizonte de planejamento, uma configuração de designação diferente pode ser gerada para cada intervalo de tempo  $h \in H$ . Além disso, também são utilizadas três matrizes de tempo adicionais, cada uma delas construída a partir de uma das seguintes informações: i) tempos máximos; ii) tempos mínimos; e iii) tempos médios de deslocamento ao longo do horizonte de planejamento. Para designação de clientes adota-se o seguinte procedimento.

Dado um intervalo  $h \in H$  e sua respectiva matriz de tempo de deslocamento, ou ainda, dada uma das matrizes de tempo adicionais recém-descritas, para cada cliente  $i \in V_c$  identifica-se o depósito com acesso mais rápido. Ele é definido com base no menor tempo de deslocamento entre esse depósito e o cliente, considerando ida e retorno. Em seguida, os clientes são listados

---

**Algoritmo 8** Estrutura geral do algoritmo meta-heurístico
 

---

```

1:  // INICIALIZAÇÃO
2:  Geração de um conjunto de soluções iniciais.
3:  enquanto a população inicial  $\leq \eta$  faça
4:    Incremento na população, a partir das soluções iniciais.
5:  fim enquanto
6:  // EVOLUÇÃO
7:  enquanto o número de iterações sem melhoria  $< \delta$  faça
8:    Seleção dos pais para a próxima geração.
9:    Aplicação de operadores de cruzamento.
10:   Aplicação de operadores de mutação direcionada.
11:   Atualização da população sobrevivente.
12:  fim enquanto
13:  // INTENSIFICAÇÃO
14:  enquanto houver melhoria em alguma solução corrente faça
15:    Aplicação de operadores de troca.
16:    Atualização da população sobrevivente.
17:  fim enquanto
18:  para cada uma das  $\theta$  melhores soluções faça
19:    Aplicação de heurística de melhoria.
20:    Atualização do custo da solução.
21:  fim para
22:  Retorna a melhor solução.
  
```

---

em ordem decrescente, de acordo com esse tempo. Partindo do topo dessa lista, designam-se os clientes aos respectivos depósitos, observando a restrição de capacidade destes. Caso uma violação ocorra, o cliente retorna para a lista e o próximo depósito de acesso mais rápido é identificado, entre os disponíveis. A lista de clientes é reordenada e o processo de designação continua. Esse procedimento é aplicado repetidamente aos clientes restantes, até que todos tenham sido alocados.

Acrescenta-se ainda um procedimento adicional à etapa de designação, que consiste em impedir a utilização de um ou mais depósitos, desnecessários para atendimento dos clientes. Ele visa gerar novas configurações, capazes de usar um número menor de instalações. Inicialmente, impossibilita-se a utilização de um dos depósitos e verifica-se a viabilidade de alocar todos os clientes nos demais, utilizando o processo de designação previamente descrito sobre a matriz de tempos médios. Sobre o conjunto de depósitos, aplica-se combinação sem repetição de elementos,  $C(|V_d|, 1)$ , para que todas as possibilidades sejam executadas. Em seguida, testa-se a viabilidade de alocação dos clientes quando dois depósitos são impossibilitados de uso, usando  $C(|V_d|, 2)$  para gerar todas as combinações possíveis.

Finalizado o processo de designação, inicia-se a construção de rotas, usando apenas os veículos de maior capacidade, de acordo com o número necessário. Duas heurísticas distintas são aplicadas para cada diferente alocação de clientes. Na primeira, as rotas são construídas de forma sequencial, ou seja, uma é finalizada para que a construção da outra inicie. Os clientes designados a um mesmo depósito são adicionados a uma lista  $L_D$ . Em seguida, um deles é

selecionado aleatoriamente, inserido ao final da rota corrente e retirado da lista  $L_D$ . O processo de seleção e inserção se repete até que a rota se torne ineficaz, ou por ultrapassar o horizonte de planejamento ou por violar a restrição de capacidade do veículo. Quando isso ocorre, o último cliente inserido é removido e retorna para a lista de não visitados  $L_D$ , e uma nova rota inicia, repetindo o processo até que todos os clientes estejam inseridos em uma das rotas. Em seguida, essa operação se estende aos demais depósitos abertos, até que a solução completa seja gerada.

A segunda heurística de geração de rotas usa o algoritmo de Clarke e Wright (1964). Contudo, ao invés de usar uma matriz de distâncias, utiliza-se a matriz de tempos que gerou a respectiva designação. Ela pode estar associada a um dos intervalos  $h \in H$  ou ser uma das matrizes adicionais. Novamente, consideram-se aqui apenas os veículos de maior capacidade, de acordo com o número necessário. Note que a matriz de tempo utilizada influencia a designação de clientes aos depósitos, e nesse caso, exerce influência também sobre a geração das rotas. O intuito da utilização de duas heurísticas distintas é diversificar a distribuição de clientes entre as rotas de cada depósito.

Concluída a geração de rotas, estas passam por uma etapa de melhoria que envolve permutação. Conforme o número de clientes por ela atendidos, utiliza-se ou a permutação de toda a sequência, ou então uma heurística que envolve permutações parciais. Nesta última, aplica-se a permutação a um subgrupo de  $v_2$  clientes, de forma sequencial e iterativa. Esse subgrupo é alterado a cada iteração, excluindo-se o primeiro cliente da subsequência e adicionando-se o seguinte, conforme ordenação da rota. Após a inclusão do último cliente a esse subgrupo, retorna-se ao seu início. O processo continua enquanto alguma melhoria for gerada após avaliação de todos os subgrupos da rota. O Algoritmo 9 descreve a etapa de melhoria intra-rota.

Depois de melhorada, cada rota é incluída a um conjunto  $\Lambda$ , com o intuito de poder recuperá-la posteriormente, caso também seja usada em outra solução. A solução completa é adicionada à população inicial  $\Gamma_i$ .

Na segunda etapa do algoritmo de inicialização, novas soluções são geradas como incremento a população  $\Gamma_i$ . Para isso, operadores de larga busca em vizinhança, que contemplam remoção e inserção de clientes, são aplicados às soluções iniciais. Eles são detalhados a seguir.

- **Operador de remoção com fechamento de depósito:** aplicado por Hemmelmayr, Cordeau e Crainic (2012) e Koç et al. (2016), este operador seleciona aleatoriamente um depósito aberto e o fecha. Todos os clientes a ele designados são transferidos para uma lista de remoção  $L_r$ . O operador também seleciona aleatoriamente um depósito fechado e o abre.
- **Operador de remoção com abertura de depósito:** este operador é uma adaptação da versão usada por Hemmelmayr, Cordeau e Crainic (2012) e Koç et al. (2016). Ele seleciona aleatoriamente um depósito fechado e o abre. Em seguida,  $n'$  clientes são removidos da solução corrente e colocados em uma lista  $L_r$ . O critério de remoção é baseado no menor tempo médio de deslocamento entre o cliente e o depósito recém-



---

**Algoritmo 9** Melhoria intra-rota
 

---

```

1: Entrada: Rota de  $n$  clientes, com tempo total de execução  $t_m$ .
2:  $Seq \leftarrow$  Sequência de clientes da rota.
3: se  $n \leq v_1$  então
4:   para cada permutação  $P_n$  da sequência original de clientes da rota faça
5:      $t_a \leftarrow$  tempo total de execução da rota, com clientes dispostos na sequência da permutação atual.
6:     se  $t_a < t_m$  então
7:        $t_m \leftarrow t_a$ 
8:        $Seq \leftarrow$  Sequência de clientes da permutação atual.
9:     fim se
10:   fim para
11: senão
12:   enquanto houver redução no tempo total de execução  $t_m$  faça
13:     para  $\varphi = 1$  até  $n$  faça
14:       se  $\varphi + v_2 \leq n$  então
15:          $SubSeq \leftarrow$  Sequência de  $v_2$  clientes da rota atual, que inicia na posição  $\varphi$ .
16:       senão
17:          $SubSeq \leftarrow$  Sequência dos  $(n - \varphi)$  últimos clientes da rota atual, sucedida pelos  $v_2 - (n - \varphi)$  primeiros clientes da rota atual.
18:       fim se
19:       para cada permutação  $P_{v_2}$  em  $SubSeq$  faça
20:          $t_a \leftarrow$  tempo total de execução da rota, com clientes de  $SubSeq$  dispostos na sequência da permutação atual.
21:         se  $t_a < t_m$  então
22:            $t_m \leftarrow t_a$ 
23:            $Seq \leftarrow$  Sequência de clientes da permutação atual.
24:         fim se
25:       fim para
26:     fim para
27:   fim enquanto
28: fim se
29: Saída: Rota de  $n$  clientes, com tempo total de execução  $\leq t_m$ .
  
```

---

aberto. Para calculá-los, considera-se o tempo de travessia nos respectivos arcos, em ambos os sentidos e para todo  $h \in H$ .

- **Operador de remoção aleatória:** usado por Ropke e Pisinger (2006) e Koç et al. (2016), este operador seleciona aleatoriamente  $n'$  clientes e os coloca em uma lista de removidos  $L_r$ .
- **Operador de remoção com fechamento de depósito melhorado:** baseado no primeiro operador de remoção, este difere no critério aplicado para escolher o depósito que será aberto. Ele foi originalmente proposto por Koç et al. (2016), mas adaptado para que esse critério considere o menor tempo médio de deslocamento entre o depósito fechado e aquele que será aberto. Como anteriormente, o cálculo considera o arco que liga ambos, nos dois sentidos e para todo  $h \in H$ .

- **Operador de inserção gulosa:** constitui uma adaptação do operador aplicado por Ropke e Pisinger (2006). Nele, os clientes são inseridos em ordem aleatória, observando a posição que minimiza o custo de inserção. Seleciona-se aleatoriamente um cliente da lista de removidos  $L_r$  e testa-se a sua inserção em todas as posições possíveis de cada uma das rotas existentes. Além disso, o custo também é avaliado para rotas novas, nos depósitos já abertos. Isso é feito observando as restrições de capacidade, tanto de depósitos quanto de veículos, de forma a manter a factibilidade. A definição da rota em que o cliente será alocado não leva em conta os custos fixos associados à utilização dos veículos.
- **Operador de inserção gulosa melhorada:** este operador é baseado na versão anterior, mas difere dela pelo fato de a inclusão não ocorre de forma aleatória. Testa-se a posição que minimiza o custo de inserção para cada um dos clientes, tal como aplicado em Koç et al. (2016). Aqui, isso é feito tanto em rotas já existentes como em novas, e sempre respeitando a factibilidade destas. Em seguida, aloca-se o cliente com menor custo de inserção na respectiva posição. Após cada inclusão, os demais clientes ainda não alocados são reavaliados. Esse processo iterativo se repete até que todos tenham sido inseridos.
- **Operador de inserção pelo maior arrependimento:** também é uma adaptação do operador aplicado por Ropke e Pisinger (2006). Informações antecipadas são usadas para evitar que clientes sejam inseridos em posições ruins porque as melhores não estão mais disponíveis. No entanto, esse mecanismo mais sofisticado também requer um número maior de cálculos. Os clientes são inseridos de acordo com o maior arrependimento, e esse valor é calculado pela diferença de custo entre inseri-lo na primeira ou na segunda melhor posição. Todas as possibilidades são testadas, tanto em rotas existentes quanto em novas, desde que se mantenham factíveis. Após uma inserção, todos os clientes da lista de remoção são reavaliados. O processo se repete até que essa lista esteja vazia.

O número de clientes removidos  $n'$  é escolhido aleatoriamente a partir de uma distribuição uniforme discreta no intervalo  $[n_l^i, n_u^i]$ . Este, por sua vez, é definido por um limite inferior e superior, calculados como percentuais do número total de clientes da instância.

Cada nova solução é gerada a partir de uma das  $\eta_0$  soluções iniciais da primeira etapa, selecionada randomicamente. Em seguida, um operador de remoção e um de inserção são escolhidos de forma aleatória, entre os recém-descritos, e aplicados sobre a solução inicial. Todas as rotas alteradas pela remoção ou inserção são avaliadas. Caso o Algoritmo 9 ainda não tenha sido aplicado a alguma delas, tal procedimento é realizado e, após isso, a rota é inserida ao conjunto  $\Lambda$ . A solução completa é adicionada à população inicial  $\Gamma_i$ . Esse processo se repete até esta tenha tamanho igual a  $\eta$ . O Algoritmo 10 representa a fase de inicialização da meta-heurística.

---

**Algoritmo 10** Inicialização
 

---

```

1:  $\eta_0 \leftarrow 0$ .
2: para todo  $h \in H$  faça
3:   Designar os clientes aos depósitos usando a matriz de tempos  $h$ .
4:   para  $\tau = 1$  até 2 faça
5:     se  $\tau = 1$  então
6:       Gerar rotas com a heurística de inserção aleatória de clientes.
7:     senão
8:       Gerar rotas usando a heurística de inserção Clarke e Wright.
9:     fim se
10:    se a solução completa é nova então
11:      SUB-ROTINA: Melhorar e guardar (solução completa)
12:       $\eta_0 \leftarrow \eta_0 + 1$ .
13:    fim se
14:  fim para
15: fim para
16: Criar as matrizes de tempo adicionais (mínimo, médio e máximo).
17: para cada matriz adicional faça
18:   Designar os clientes aos depósitos de acordo com o tempo de deslocamento dessa matriz.
19:   Gerar rotas usando a heurística de inserção Clarke e Wright.
20:   se a solução completa é nova então
21:     SUB-ROTINA: Melhorar e guardar (solução completa)
22:      $\eta_0 \leftarrow \eta_0 + 1$ .
23:   fim se
24: fim para
25: para cada  $C(|V_d|, 1)$  ou  $C(|V_d|, 2)$  faça
26:   Desative o(s) depósito(s) correspondente(s).
27:   Designar os clientes aos depósitos disponíveis, considerando a matriz de tempos médios.
28:   se for possível designar todos os clientes então
29:     Gerar rotas usando a heurística de inserção Clarke e Wright.
30:     se a solução completa é nova então
31:       SUB-ROTINA: Melhorar e guardar (solução completa)
32:        $\eta_0 \leftarrow \eta_0 + 1$ .
33:     fim se
34:   fim se
35: fim para
36: enquanto  $\eta_1 \leq \eta$  faça
37:   Selecionar aleatoriamente uma das  $\eta_0$  soluções iniciais.
38:   Selecionar aleatoriamente um operador de remoção e aplicá-lo nessa solução inicial.
39:   Colocar os  $n'$  clientes removidos em uma lista  $L_r$ .
40:   Selecionar aleatoriamente um operador de inserção e aplicá-lo aos clientes da lista  $L_r$ .
41:   se a solução completa gerada é nova então
42:     SUB-ROTINA: Melhorar e guardar (solução completa)
43:      $\eta_1 \leftarrow \eta_1 + 1$ .
44:   fim se
45: fim enquanto
46: SUB-ROTINA: Melhorar e guardar (solução completa)
47:   Melhorar as rotas inéditas através do Algoritmo 9.
48:   Salvar cada rota nova no conjunto  $\Lambda$  e a solução completa na população inicial  $\Gamma_i$ .

```

---

#### 4.2.2 Evolução

Esta fase envolve o processo de geração de novas soluções, que inclui operadores de cruzamento e mutação. Este último é aliado a um procedimento de melhoria heurística, que considera o contexto da frota heterogênea. O intuito é diversificar os locais de busca e melhorar a qualidade do conjunto de soluções criadas a cada iteração. A seguir, detalham-se as estratégias usadas nessa fase.

Na geração de soluções por cruzamento, aplica-se o operador *Partially Mapped Crossover* (GOLDBERG; LINGLE, 1985). Ele utiliza dois pontos de corte e preserva a posição vinda dos pais. Inicialmente, selecionam-se aleatoriamente duas soluções (pais),  $P_1$  e  $P_2$ . Para cada uma delas, os clientes são colocados em uma sequência única, aqui definida de acordo com a formação das respectivas rotas. Em seguida, determinam-se os dois pontos de corte para essas sequências, que indicam quantos genes serão cruzados. Eles são obtidos fixando o número de clientes que será removido. Esse número é selecionado aleatoriamente a partir de uma distribuição uniforme discreta no intervalo  $[n_l^c, n_u^c]$ , definido por limites inferior e superior que são calculados como percentuais do total de clientes da instância. Logo após, determina-se a posição da sequência onde o primeiro ponto de corte inicia. Isso é feito de forma aleatória, entre uma das posições viáveis. Os genes contidos entre esses dois pontos são então cruzados para geração dos descendentes. Entretanto, durante a troca de material genético, é possível que os cromossomos passem a ter genes repetidos. Neste caso, substitui-se todos aqueles que estão fora da região de corte pelo gene pertencente ao mesmo *locus*, mas no outro cromossomo. Gerada a sequência de clientes de cada descendente, esses são distribuídos novamente em rotas, conforme a configuração de um dos pais, respeitando a ordem de distribuição da sequência e o número de clientes das respectivas rotas. Por fim, as novas soluções geradas são testadas e, caso alguma rota seja infactível, a solução que a contém é descartada.

Na geração de soluções por mutação, cria-se um descendente a partir da cópia de uma outra solução, que sofre alterações em um ou mais genes do cromossomo. Com o intuito de melhorar a qualidade a cada nova geração, essa mutação é direcionada, ou seja, ela envolve uma melhoria heurística. Além disso, as soluções geradas com os operadores de mutação são sempre factíveis. O procedimento é composto por duas etapas: remoção, seguida de inserção. Na primeira,  $n'$  clientes são iterativamente alocados em uma lista de remoção  $L_r$  por um dos operadores dedicados a essa fase. Novamente,  $n'$  é um número selecionado aleatoriamente a partir de uma distribuição uniforme discreta no intervalo  $[n_l^m, n_u^m]$ , definida por um limite inferior e superior, calculados como percentuais do total de clientes da instância. Durante a etapa de inserção, os clientes removidos são realocados na solução incompleta. Ambos os operadores são escolhidos aleatoriamente entre o conjunto disponível, que é detalhado a seguir.

- **Operador de remoção – extremos:** aplicado em diferentes trabalhos Ropke e Pisinger (2006), Demir, Bektaş e Laporte (2012), Koç et al. (2016), e aqui adaptado para o problema em questão, este operador remove os  $n'$  clientes que são extremos em relação à

rota a qual pertencem. Para identificá-los, calcula-se o tempo necessário para chegar até ele, vindo do seu antecessor, somado ao tempo necessário para ir dele até o seu sucessor, de acordo com o intervalo  $h$  correspondente ao momento de travessia desses arcos.

- **Operador de remoção – custos extremos:** este operador remove os  $n'$  clientes que são extremos em relação ao custo de inserção na rota. Quando um cliente é adicionado, isso não só altera o custo associado com os arcos modificados, mas também pode alterar os custos nos arcos subsequentes, devido à possibilidade de mudança no intervalo de tempo a eles associado. O custo de inserção é calculado pela diferença entre o tempo de execução da rota com o cliente e sem ele. Uma versão deste operador baseada em distância foi aplicada por Ropke e Pisinger (2006), Demir, Bektaş e Laporte (2012) e Koç et al. (2016).
- **Operador de remoção – depósito menos eficiente:** proposto por Koç et al. (2016), este operador verifica a taxa de utilização de cada depósito aberto, dada pela relação entre a soma das demandas dos clientes a ele designados e sua capacidade. O depósito com a menor taxa de utilização é fechado e os seus  $n'$  clientes alocados na lista de remoção  $L_r$ .
- **Operador de remoção – veículo menos eficiente:** semelhante a ideia anterior, propõe-se um operador que verifica a taxa de utilização de cada veículo vinculado a uma rota. Essa taxa é dada pela relação entre demanda total dos clientes por ele atendidos e sua capacidade. A rota associada ao veículo com a menor taxa de utilização é removida, e os seus  $n'$  clientes alocados na lista de remoção  $L_r$ .
- **Operador de inserção – menos custoso:** este é um operador de inserção gulosa que aloca os clientes iterativamente nas rotas, na posição que minimiza o custo de inserção. Diferente dos operadores da fase de inicialização, aqui consideram-se também os custos fixos de utilização dos veículos e abertura dos depósitos. Dada uma solução, após a remoção dos clientes, verifica-se se aqueles que restaram em cada rota podem ser atendidos por um veículo menor e, também, se há algum depósito aberto que ficou sem designação de clientes. Em caso positivo, o custo é atualizado. Em seguida, computa-se o custo de inserção de cada um dos clientes removidos, considerando, inclusive, os custos fixos adicionais que essa inclusão pode ocasionar. A cada iteração, obtém-se o menor custo de inserção para cada cliente e, insere-se aquele que for menos custoso. O processo de reavaliação é feito a cada iteração até que todos os clientes removidos tenham sido reinseridos.
- **Operador de inserção – mais custoso:** este operador é semelhante ao anterior, diferindo apenas pelo fato que, a cada iteração, após calcular o menor custo de inclusão de cada cliente, insere-se aquele cliente que, entre os menores custos de inclusão, apresenta o maior valor.

Na sequência, descrevem-se os detalhes do processo iterativo de evolução e seleção das soluções.

Com uma população inicial  $\Gamma_i$ , de tamanho  $\eta$ , selecionam-se dela duas soluções de forma aleatória,  $P_1$  e  $P_2$ , e aplica-se o operador de cruzamento. Se algum dos dois indivíduos criados for factível e inédito, a solução é computada e adicionada à geração corrente  $\Gamma_g$ . Esse processo se repete até que  $\delta_1$  novos indivíduos sejam criados. Em seguida, o procedimento de mutação direcionada é aplicado iterativamente às soluções da população inicial  $\Gamma_i$ . Para cada uma delas, um operador de remoção e outro de inserção são escolhidos aleatoriamente, entre os disponíveis. Depois de empregados, verifica-se se a solução criada é inédita ou não. Em caso positivo, ela é adicionada à geração corrente  $\Gamma_g$ , e as rotas novas são inseridas no conjunto  $\Lambda$ . Esse procedimento é aplicado sequencialmente a toda população inicial, ou até que  $\eta$  novos indivíduos sejam criados. Na sequência,  $\Gamma_i$  e  $\Gamma_g$  são agrupados para compor a população atual  $\Gamma$ . Esta é então ordenada conforme o custo, do menor para o maior.

Assim, após a primeira geração, a população atual  $\Gamma$  pode conter até  $2\eta$  soluções ordenadas. Entre elas, uma parte é usada para criar os descendentes. Esse subconjunto  $\Gamma_p$  exerce o papel de pai, da mesma forma que a população inicial exerceu na primeira geração. O critério para escolha dos pais é o seguinte. Diante da população atual  $\Gamma$ , selecionam-se as  $\chi_1$  melhores soluções, as  $\chi_3$  piores e, entre as restantes, elegem-se mais  $\chi_2$  soluções aleatoriamente, de forma que  $\chi_1 + \chi_2 + \chi_3 = \eta$ . Os operadores de cruzamento e mutação direcionada são então aplicados à  $\Gamma_p$ , conforme descrito previamente. Na seleção da população sobrevivente, mantêm-se apenas as  $\eta$  melhores soluções de  $\Gamma$ , e as demais são descartadas. A essa população atual  $\Gamma$ , acrescenta-se, em seguida, a nova geração  $\Gamma_g$ .

Os indivíduos sobreviventes são então ordenados pelo menor custo e o processo de seleção dos pais e criação de uma nova geração se repete. O processo encerra quando não houver mais melhoria em nenhuma das  $\sigma$  soluções do topo, durante  $\delta$  iterações consecutivas.

A fase de evolução é descrita através do Algoritmo 11. Encerrada essa fase, a população sobrevivente é novamente ordenada, e as  $\eta$  melhores soluções são armazenadas para a fase posterior.

#### 4.2.3 Intensificação

A última fase do algoritmo meta-heurístico visa intensificar a busca nas soluções remanescentes. Ela é dividida em duas principais etapas. A primeira abrange um processo iterativo, semelhante ao que foi aplicado na fase de evolução. Ele envolve dois operadores que executam trocas inter-rotas, um relacionado com as demandas dos clientes e outro com tempo de serviço. A segunda etapa compreende uma heurística que alia o melhor momento de saída de cada veículo com a sequência de clientes mais adequada para esse momento. Note aqui que o grupo de clientes atendido pelo veículo não é alterado, apenas sua sequência, caso essa mudança provoque redução de custos.

---

**Algoritmo 11** Evolução
 

---

```

 $\Gamma_p \leftarrow \emptyset; \Gamma_g \leftarrow \emptyset$ 
 $\Gamma_p \leftarrow \Gamma_i$  (A população inicial é utilizada para criar a próxima geração)
 $\Gamma \leftarrow \Gamma_i$ 
enquanto número de iterações sem melhoria nas  $\sigma$  melhores soluções de  $\Gamma \leq \delta$  faça
   $\tau \leftarrow 0$ 
  enquanto o número de soluções factíveis e inéditas  $\leq \delta_1$  faça
    Selecionar aleatoriamente  $P_1$  e  $P_2$  em  $\Gamma_p$ .
    Aplicar o operador de cruzamento.
    para cada solução factível e inédita gerada faça
      Melhorar as rotas através do Algoritmo 9.
      Incluir a solução em  $\Gamma_g$ .
       $\tau = \tau + 1$ 
      se  $\tau = \delta_1$  então
        Interromper laço
      fim se
    fim para
  fim enquanto
  para cada solução em  $\Gamma_p$  faça
    Sortear aleatoriamente um operador de remoção da lista e aplicá-lo à solução.
    Colocar os  $n'$  clientes removidos em uma lista  $L_r$ .
    Selecionar aleatoriamente um operador de remoção e aplicá-los aos clientes da lista  $L_r$ .
    se a solução completa gerada é nova então
      Melhorar as rotas através do Algoritmo 9.
      Incluir a solução em  $\Gamma_g$ .
    fim se
  fim para
  Ordenar  $\Gamma$  pelo menor custo e manter apenas as  $\eta$  melhores soluções.
   $\Gamma \leftarrow \Gamma_g; \Gamma_g \leftarrow \emptyset$ .
  Ordenar  $\Gamma$  pelo menor custo.
   $\Gamma_p$  recebe um subconjunto de  $\Gamma$ , conforme especificações de  $\chi_1 + \chi_2 + \chi_3$ .
fim enquanto

```

---

A seguir são apresentados os detalhes dos operadores da primeira etapa e explica-se como eles são aplicados no algoritmo.

- **Operador de troca baseado em demanda:** aplicado por diferentes autores (ROPKE; PISINGER, 2006; DEMIR; BEKTAŞ; LAPORTE, 2012; KOÇ et al., 2016), este operador seleciona dois clientes de rotas distintas, semelhantes em relação à demanda, e executa a troca entre eles. Aqui, isso só é realizado caso essa troca não gere infactibilidade. Em seguida, as rotas a que esses clientes pertencem são testadas e, se possível melhoradas, através do Algoritmo 9.
- **Operador de troca baseado em tempo de serviço:** este operador é semelhante ao anterior, diferindo apenas pelo fato de que os clientes são escolhidos pela semelhança em relação ao tempo de serviço que demandam.

Após a aplicação de um desses operadores, verifica-se se a alteração produziu uma solução inédita ou não. Em caso positivo, ela é inserida na população  $\Gamma$  e as rotas adicionadas ao conjunto  $\Lambda$ . Do contrário, é descartada.

De forma mais precisa, o processo iterativo é executado conforme segue. Primeiro, o operador de troca baseado em demanda é aplicado sobre a população sobrevivente da fase anterior  $\Gamma$ . As soluções por ele alteradas são incorporadas a população como novos indivíduos. Ao final, todas são ordenadas pelo menor custo, mantendo-se na população apenas as  $\eta$  melhores soluções. Na sequência, o operador de troca baseado em tempo de serviço é aplicado na população remanescente. Da mesma forma, ao final, as soluções são ordenadas e mantêm-se apenas as  $\eta$  melhores. Essa sequência de procedimentos é efetuada repetidamente, até que nenhuma nova melhoria tenha sido produzida com as trocas.

Encerrada a primeira etapa, aplica-se uma heurística de melhoria que busca aprimorar as soluções do topo, aliando o momento da saída com a sequência de ordenação dos clientes. Ela é aplicada nos indivíduos que compõem as  $\theta$  melhores soluções, e funciona da seguinte forma. Cada rota que compõe uma solução é tratada individualmente. Parte-se da sequência de clientes salva durante a sua geração, considerando a partida no início do horizonte de planejamento. O tempo total de execução da rota considera a saída do veículo do depósito ao qual está vinculada até seu retorno nesse mesmo ponto. O processo é cíclico, e a cada nova iteração incrementa-se  $\Delta_t$  unidades de tempo ao momento de saída no depósito. Caso o período  $h \in H$  de atendimento em algum cliente tenha sido alterado com esse incremento, aplica-se o Algoritmo 9 visando a melhoria intra-rota. Se, após isso, o tempo total de execução nessa rota é menor, substitui-se a sequência de clientes original pela nova e, altera-se o momento de saída. Esse processo é repetido enquanto o incremento adicionado não gerar infactibilidade à execução da rota. Ao final desse ciclo, obtém-se o melhor momento de saída e a sequência de clientes que gerou o menor tempo de execução da rota. Estende-se esse mesmo procedimento às demais rotas e, caso alguma melhoria seja realizada, a solução original é substituída pela melhorada. Após a aplicação dessa heurística a todas as soluções do topo, estas são reordenadas de acordo com o menor custo, para identificação da melhor solução.

A fase de intensificação das soluções, descrita através do Algoritmo 12, encerra o algoritmo meta-heurístico.

### 4.3 TESTES COMPUTACIONAIS

Nesta seção, descrevem-se como foram conduzidos os testes computacionais e definidos os parâmetros utilizados nos algoritmos que compõem a abordagem meta-heurística, além das informações usadas na geração das instâncias de teste.

Os algoritmos foram implementados em linguagem C++. Usou-se o *solver* Gurobi Optimizer 8.1.1 para resolver os modelos de programação inteira mista, na configuração padrão. Todos os testes computacionais foram conduzidos em um processador Intel Core i7, rodando a



---

**Algoritmo 12** Intensificação
 

---

```

enquanto número de iterações sem melhoria  $\leq 0$  faça
  para cada solução em  $\Gamma$  faça
    Aplicar o operador de troca baseado em demanda
    se solução gerada é inédita então
      Melhorar as rotas através do Algoritmo 9.
      Incluir a solução em  $\Gamma$ .
    fim se
  fim para
  Ordenar  $\Gamma$  pelo menor custo e manter apenas as  $\eta$  melhores soluções.
  para cada solução em  $\Gamma$  faça
    Aplicar o operador de troca baseado em tempo de serviço
    se solução gerada é inédita então
      Melhorar as rotas através do Algoritmo 9.
      Incluir a solução em  $\Gamma$ .
    fim se
  fim para
  Ordenar  $\Gamma$  pelo menor custo e manter apenas as  $\eta$  melhores soluções.
fim enquanto
para cada uma das  $\theta$  soluções do topo faça
  para cada rota da solução faça
     $\tau_0 \leftarrow$  tempo de execução rota, considerando o momento de saída  $t_s = 0$ .
     $Seq \leftarrow$  Sequência de clientes da rota
    enquanto  $(t_s + \Delta_t + \sum_{i \in V_c} s_i) \leq |H|\bar{T}$  faça
       $t_s = t_s + \Delta_t$ .
      Aplicar Algoritmo 9, considerando saída no instante  $t_s$ .
       $\tau_1 =$  tempo de execução da rota para saída no instante  $t_s$ .
      se  $\tau_1 < \tau_0$  então
         $\tau_0 \leftarrow \tau_1$ .
         $Seq \leftarrow$  Sequência atualizada de clientes da rota.
      fim se
    fim enquanto
    Atualizar a sequência da rota para  $Seq$  e o tempo de execução para  $\tau_0$ .
  fim para
  Atualizar o custo da solução completa.
fim para
  Retorna a melhor solução completa.
  
```

---

3.4 GHz, com 64 GB de memória RAM instalada, no sistema operacional Ubuntu Linux, com compilador gcc 7.4.0. Dois *threads* foram fixados para utilização do *solver* e um limite total de 10800 segundos foi imposto na execução de cada configuração de teste.

O método foi ajustado através da realização de testes preliminares, para os quais diferentes valores foram testados em cada um dos parâmetros que compõem o algoritmo. Os melhores resultados médios foram obtidos com os parâmetros fixados conforme especificado a seguir. O tamanho da população inicial  $\eta$  foi definido em função do número de clientes da instância, de tal forma que  $\eta = 30\sqrt{|V_c|}$ . Definiu-se essa relação para evitar que, para instâncias com um número maior de clientes, a população de cada geração se tornasse muito grande. Na

heurística de melhoria intra-rota,  $v_1$  também foi fixado de acordo com o número de clientes das instâncias. Esse parâmetro foi definido em 10, 8, 7 e 6, respectivamente, para instâncias com 10, 20, 50 e 80 clientes ou mais;  $v_2$  foi fixado em 3 para todas as configurações. Na composição de  $\Gamma_p$  estabeleceu-se  $\chi_1 = 0,5\eta$ ,  $\chi_2 = 0,4\eta$  e  $\chi_3 = 0,1\eta$ . Para a fase de inicialização, o intervalo ao qual pertence o número de clientes removidos  $n'$  foi definido por  $n_l^i = 0,3|V_c|$  e  $n_u^i = 0,6|V_c|$ . Na fase de evolução, os limites do intervalo considerado no operador de cruzamento foram fixados em  $n_l^c = 0,2|V_c|$  e  $n_u^c = 0,4|V_c|$ , e o número de indivíduos criados a cada iteração como  $\delta_1 = 0,02\eta$ . Para o operador de mutação, fixou-se os limites em  $n_l^m = 0,2|V_c|$  e  $n_u^m = 0,5|V_c|$ . Estabeleceu-se o número de soluções da elite em função da população inicial, de tal forma que  $\sigma = 0,1\eta$  para instâncias com 10 e 20 clientes,  $\sigma = 0,15\eta$  para instâncias com 50 clientes e  $\sigma = 0,2\eta$  para as demais. Por fim, para a fase de evolução, também relacionou-se o número de soluções da elite com  $\eta$ , de modo que  $\vartheta = 0,05\eta$  em todas as configurações. Para que todos os parâmetros assumissem valores inteiros, os resultados foram arredondados, quando necessário, para o inteiro positivo mais próximo.

As dez melhores soluções obtidas na etapa de inicialização do algoritmo foram guardadas para servir de entrada (solução inicial) para o modelo de programação inteira mista apresentado na Seção 4.1. Contudo, as soluções fornecidas ao *solver* foram apenas parciais. Informou-se unicamente a sequência de arcos atravessados, deixando que o momento de saída fosse pré-processado e definido pelo próprio *solver*, antes do início da resolução do modelo.

#### 4.3.1 Instâncias de teste

Parte do conjunto de instâncias descrito na Seção 3.3.1 foi adaptado para a condução dos testes computacionais relativos ao PLRDFH-DT. Excluíram-se apenas as instâncias com um único depósito. Para tratar do problema em questão, o número de veículos foi substituído pela quantidade de diferentes tipos existentes na frota. Incluíram-se a capacidade e o custo de abertura dos potenciais depósitos, a capacidade e o custo fixo de utilização de cada tipo de veículo, e também o custo por unidade de tempo para a etapa de roteamento. As demais informações foram mantidas conforme as instâncias originais.

O número de diferentes tipos de veículos na frota,  $|K|$ , foi definido em 3 para todas as instâncias. A especificação de capacidade dos mesmos  $Q_k, \forall k \in K$  foi fixada em 2000, 4000 e 6000 unidades, e o custo fixo  $F_k$  de utilização em 1000, 1500 e 2000 unidades monetárias, respectivamente. A capacidade de cada depósito  $W_i, \forall i \in V_d$  foi gerada aleatoriamente a partir de uma distribuição uniforme discreta no intervalo  $[w_l^i, w_u^i]$ . Os limites inferior e superior são definidos por percentuais da demanda total dos clientes, fixados em 50% e 85%, respectivamente. O custo de abertura de cada potencial depósito é proporcional a sua capacidade. Para um depósito que comporte exatamente a demanda total dos clientes, fixou-se o custo em 13500 unidades monetárias. Todos os valores foram arredondados para o inteiro mais próximo, quando necessário. Para cada unidade de tempo vinculada à etapa de roteamento, o custo foi definido em 1 unidade monetária, indiferente do tipo de veículo usado.

Assim, cada instância foi identificada pelo número de intervalos de tempo ( $|H|$ ), de potenciais depósitos ( $|V_d|$ ) e de clientes ( $|V_c|$ ). A Tabela 4.2 resume as diferentes configurações geradas.

Tabela 4.2: CONJUNTOS DE INSTÂNCIAS PARA O PLRDFH-DT E SEUS PARÂMETROS

Tipo	$ H $	$\bar{T}$ (s)	$ V_d $	$ V_c $
Pequena	5	10800	{3, 5}	10
			{3, 5}	20
			{3, 5}	50
			{3, 5}	80
			{3, 5}	100
Média	10	5400	{3, 5}	10
			{3, 5}	20
			{3, 5}	50
			{3, 5}	80
			{3, 5}	100
Grande	15	3600	{3, 5}	10
			{3, 5}	20
			{3, 5}	50
			{3, 5}	80
			{3, 5}	100

FONTE: A autora (2019)

#### 4.4 RESULTADOS E DISCUSSÃO

Nesta seção são apresentados e discutidos os resultados alcançados com a realização de intensivos testes computacionais. Primeiro, apresentam-se os resultados da formulação matemática descrita na Seção 4.1, acompanhada de um conjunto de soluções iniciais. Em seguida, reportam-se os dados gerados quando essa configuração inicial é fortalecida por um conjunto de desigualdades válidas. Realiza-se ainda uma análise da eficiência dessa inclusão. Resultados da aplicação da meta-heurística, descrita na Seção 4.2, são detalhados na sequência e comparados aos obtidos pelo modelo de programação linear inteira mista. Ao final, complementa-se esta seção com uma avaliação do desempenho do algoritmo meta-heurístico na resolução do PLR-DT, proposto na Seção 3.1. Contrastam-se os resultados obtidos através dele com aqueles gerados pelo algoritmo metaheurístico descrito na Seção 3.2.

##### 4.4.1 Resultados da formulação matemática

Inicialmente, são apresentados os resultados gerados pelo modelo matemático proposto para o PLRDFH-DT, definido por (4.1) – (4.39), quando um conjunto de soluções iniciais é fornecido como entrada. A Tabela 4.3 mostra os valores médios obtidos, relativos às 5 instâncias aleatórias geradas em cada configuração de teste, considerando um limite de tempo de execução de 10800 segundos. As duas primeiras colunas fornecem informações sobre as instâncias. Na sequência, são reportados os resultados médios para os casos com 3 e 5 depósitos, respectivamente.

Para cada um desses casos, constam informações sobre a melhor solução inicial aceita pelo *solver*, a melhor solução alcançada com o modelo, ou limite superior (LS), o limite inferior (LI), o *gap* entre eles (em %), calculado como  $100(LS - LI)/LS$ , o tempo de execução (em segundos) e, finalmente, a melhoria alcançada em relação a solução inicial (em %).

Conforme os resultados da Tabela 4.3, o *solver* Gurobi não foi capaz de melhorar significativamente as soluções iniciais, mesmo após 3 horas de processamento. Uma melhoria média de apenas 0,34% foi alcançada nas instâncias com 3 depósitos e 0,64% nos casos onde há 5 potenciais depósitos. Observa-se uma diferença bastante elevada entre os limites superior e inferior, mesmo em configurações com um número menor de clientes. Isso produziu *gaps* médios de 73,63% e 74,07%, respectivamente, para instâncias com 3 e 5 depósitos. Testes preliminares mostraram ainda que para a grande maioria dos casos, nas condições estabelecidas, o *solver* não foi capaz de encontrar solução factível sem a inclusão de soluções iniciais.

A seguir, é realizada uma análise sobre a eficiência da inclusão das desigualdades válidas. Avalia-se a situação onde estas são adicionadas ao modelo matemático proposto para o PLRDFH-DT, juntamente com o conjunto de soluções iniciais. A Tabela 4.4 apresenta os resultados médios obtidos para instâncias com 3 e 5 potenciais depósitos, respectivamente. Novamente, os resultados reportam a média das 5 instâncias testadas em cada configuração, para um tempo limite de execução de 10800 segundos. As duas primeiras colunas trazem informações a respeito da instância. Em seguida, de acordo com o número de depósitos e de intervalos de tempo, reportam-se os resultados da melhor solução obtida com o modelo, ou limite superior, o limite inferior, o *gap* entre eles (em %) e o tempo de execução (em segundos). Por último, constam ainda o *gap* para os limites superior e inferior (em %), respectivamente, em relação aos resultados da Tabela 4.3, onde não houve inclusão de desigualdades válidas. Para o limite superior, ele é calculado como  $100(LS_a - LS_c)/LS_a$ , onde  $LS_a$  indica o limite superior da formulação atual e  $LS_c$  indica o respectivo limite superior da formulação completa apresentada na Tabela 4.3. O mesmo procedimento é aplicado para o limite inferior. Valores positivos para o *gap* LS indicam a melhoria gerada com essa inclusão, enquanto valores negativos traduzem a deterioração dos resultados, em comparação com a situação anterior. Já para o *gap* LI, a situação é oposta, de forma que valores negativos indicam a elevação no limite inferior em relação aos resultados anteriores.

As informações da Tabela 4.4 mostram que a inclusão das desigualdades válidas não exerce um efeito significativo sobre o limite superior. Valores equivalentes aos reportados na Tabela 4.3 são obtidos para instâncias com 3 potenciais depósitos, que apresentam melhoria média de apenas 0,73%. Em composições com 5 depósitos, essa melhoria reduz para 0,28%. Contudo, a incorporação das desigualdades válidas melhora consideravelmente os limites inferiores e, consequentemente, reduz o *gap*. A elevação média nos limites inferiores, em relação aos valores da tabela anterior, é de 198,25% nas instâncias com 3 depósitos e 194,91% naquelas onde há 5 potenciais depósitos. Para o *gap*, observam-se valores médios de 20,62% e 23,48% nas instâncias de 3 e 5 depósitos, respectivamente. Isso destaca a importância das desigualdades

Tabela 4.3: RESULTADOS MÉDIOS DO PLRDFH-DT COM SOLUÇÕES INICIAIS

Tipo	$ V_c $	3 depósitos						5 depósitos					
		Melhor SI	Melhor Solução	Limite Inferior	Gap (%)	Tempo (s)	Melhoria sobre SI (%)	Melhor SI	Melhor Solução	Limite Inferior	Gap (%)	Tempo (s)	Melhoria sobre SI (%)
Pequena	10	27143,60	26924,60	8608,05	67,89	10800	0,73	26566,80	26344,20	8712,55	66,83	10800	0,82
	20	33588,60	33344,20	10098,82	69,70	10800	0,72	33395,80	32774,00	9421,37	71,26	10800	1,85
	50	54976,60	54807,00	14500,14	73,50	10800	0,28	57926,60	57881,20	14394,92	75,09	10800	0,08
	80	72014,60	70987,60	16590,76	76,55	10800	1,51	71569,60	71397,40	17118,30	75,95	10800	0,24
	100	83410,40	83356,80	19764,08	76,28	10800	0,06	81363,20	81156,00	17540,72	78,63	10800	0,27
Média		54226,76	53884,04	13912,37	72,78	10800	0,66	54164,40	53910,56	13437,57	73,55	10800	0,65
Média	10	26956,00	26896,20	8082,97	69,88	10800	0,22	25361,40	25336,80	8211,66	67,71	10800	0,10
	20	34334,00	34055,80	10090,80	70,35	10800	0,76	33282,00	33078,00	9426,67	71,48	10800	0,64
	50	56714,60	56691,20	14317,82	74,73	10800	0,04	55116,20	54756,40	14243,34	73,90	10800	0,71
	80	72899,20	72734,00	16333,28	77,47	10800	0,21	71723,60	71551,80	16931,24	76,34	10800	0,24
	100	81308,80	81290,20	19586,20	75,91	10800	0,02	83363,80	83354,40	17326,74	79,43	10800	0,01
Média		54442,52	54333,48	13682,21	73,67	10800	0,25	53769,40	53615,48	13227,93	73,77	10800	0,34
Grande	10	27743,00	27736,40	7646,67	72,29	10800	0,03	26349,80	26302,00	7788,35	70,47	10800	0,17
	20	35414,40	35220,60	9959,38	71,67	10800	0,51	33424,00	32864,60	9286,57	71,70	10800	1,60
	50	56351,00	56336,80	14234,62	74,74	10800	0,03	59001,40	57663,80	14142,80	75,41	10800	2,20
	80	71698,20	71664,20	16181,74	77,38	10800	0,04	77251,40	76986,40	16965,22	77,90	10800	0,34
	100	82098,80	82097,20	19575,04	76,16	10800	0,00	81470,20	81141,00	17355,12	78,87	10800	0,38
Média		54661,08	54611,04	13519,49	74,45	10800	0,12	55499,36	54991,56	13107,61	74,87	10800	0,94
Média global		54443,45	54276,19	13704,69	73,63	10800	0,34	54477,72	54172,53	13257,71	74,07	10800	0,64

FONTE: A autora (2019)

Tabela 4.4: RESULTADOS MÉDIOS DO PLRDFH-DT COM SOLUÇÕES INICIAIS E DESIGUALDADES VÁLIDAS

Tipo	V <sub>c</sub>	3 depósitos						5 depósitos					
		Limite Superior	Limite Inferior	Gap (%)	Tempo (s)	Em relação a versão sem DVs		Limite Superior	Limite Inferior	Gap (%)	Tempo (s)	Em relação a versão sem DVs	
						Gap LS (%)	Gap LI (%)					Gap LS (%)	Gap LI (%)
Pequena	10	26903,00	26902,24	0,00	577	0,08	-213,95	26304,80	26143,90	0,63	3638	0,15	-204,67
	20	33258,60	31019,50	6,69	10800	0,26	-207,09	32693,00	29526,48	9,70	10800	0,25	-214,45
	50	53538,00	40297,70	24,59	10800	2,30	-177,94	57567,80	39422,30	31,35	10800	0,52	-174,93
	80	71891,40	48142,18	33,04	10800	-1,46	-189,92	70968,80	45804,90	35,35	10800	0,61	-168,18
	100	83410,40	55376,12	33,60	10800	-0,06	-181,08	80927,00	50116,96	37,58	10800	0,24	-195,93
Média		53800,28	40347,55	19,58	8755	0,22	-194,00	53692,28	38202,91	22,92	9368	0,35	-191,63
Média	10	26786,20	26259,70	1,94	4476	0,41	-225,93	25232,20	24743,74	1,94	8485	0,41	-204,88
	20	33564,80	29612,22	11,75	10800	1,43	-193,58	32569,20	28324,92	12,62	10800	1,57	-202,20
	50	54784,40	41186,16	24,61	10800	3,27	-188,59	54425,20	38958,36	28,05	10800	0,64	-173,74
	80	72734,00	47455,32	34,68	10800	0,00	-190,78	71669,20	46814,48	34,61	10800	-0,17	-177,86
	100	81278,40	54959,68	32,40	10800	0,02	-181,10	83137,40	50879,14	38,33	10800	0,25	-205,45
Média		53829,56	39894,62	21,08	9535	1,02	-195,99	53406,64	37944,13	23,11	10337	0,54	-192,82
Grande	10	27468,60	27266,90	0,67	3277	0,93	-257,33	26173,20	25966,62	0,74	5060	0,49	-237,65
	20	34062,40	29848,58	12,50	10800	3,11	-199,92	32813,20	28658,00	12,38	10800	0,21	-209,39
	50	56062,40	40873,74	27,04	10800	0,46	-188,08	58634,60	38683,46	33,84	10800	-1,88	-173,61
	80	71665,20	47456,20	33,73	10800	0,00	-193,43	76806,80	47580,58	37,86	10800	0,26	-181,39
	100	81907,80	55538,20	32,13	10800	0,23	-185,02	80608,60	50083,80	37,25	10800	0,69	-199,36
Média		54233,28	40196,72	21,21	9295	0,95	-204,76	55007,28	38194,49	24,42	9652	-0,05	-200,28
Média global		53954,37	40146,30	20,62	9195	0,73	-198,25	54035,40	38113,84	23,48	9786	0,28	-194,91

FONTE: A autora (2019)

válidas para a melhoria dos limites inferiores. Resultados mais detalhados mostram ainda que, em configurações com 10 clientes, o *solver* foi capaz de provar a otimalidade em 21 das 30 instâncias testadas, e que nestas o tempo médio para encontrar a solução ótima foi inferior a 30 minutos.

No entanto, fazendo uma análise global, percebe-se que mesmo fornecendo um grupo de soluções iniciais e fortalecendo o modelo com um conjunto de desigualdades válidas, o *gap* médio obtido continua elevado após 3 horas de execução. Isso traduz a complexidade ainda presente na resolução deste problema.

#### 4.4.2 Resultados da abordagem meta-heurística

A Tabela 4.5 detalha os resultados médios produzidos pelo algoritmo meta-heurístico proposto e os compara aos obtidos com a formulação matemática, acompanhada pelas soluções iniciais e desigualdades válidas. Ambas as configurações, com 3 e 5 potenciais depósitos, são apresentadas nesta tabela e, em cada resultado, é reportada a média das 5 instâncias correspondentes testadas. Assim como nos testes anteriores, um limite de 10800 segundos foi imposto na execução. Informações gerais sobre as instâncias são mostradas nas duas primeiras colunas. Na sequência, para cada configuração, apresentam-se os melhores resultados médios gerados em cada etapa da meta-heurística (inicialização, evolução e intensificação), seguido pelo tempo total de execução (em segundos). As duas colunas posteriores reportam a melhoria percentual obtida quando essa abordagem é comparada à prévia. Primeiro, no que tange à solução gerada, e depois em relação ao tempo de execução.

A análise dos resultados da Tabela 4.5 fornece observações relevantes. Primeiro, percebe-se como cada fase da meta-heurística contribui para o avanço na melhoria das soluções. Da primeira para a segunda fase, consegue-se elevar a qualidade média das soluções em 9,98% e 14,53%, respectivamente, para instâncias com 3 e 5 depósitos. Já da segunda para a terceira fase essa melhoria é mais sutil, representando, nesta mesma ordem, 0,87% e 0,90%. Além disso, os dados mostram que essa abordagem produz melhores soluções médias em todas as configurações, exceto para instâncias com 10 clientes. Nesses casos, os resultados médios são 0,16% piores quando envolvem 3 depósitos e 0,06% piores para 5 depósitos. Em compensação, ao comparar as soluções produzidas pelo algoritmo meta-heurístico com aquelas geradas pela formulação matemática, fortalecida pelas desigualdades válidas, observam-se melhorias que podem chegar a mais de 21%, em alguns casos. Ganhos de qualidade superiores a 9% são observados em todas as configurações com 50 clientes ou mais.

Em relação ao tempo de execução, os resultados mostram que a abordagem meta-heurística é capaz de obter soluções mais rapidamente em todas as situações, quando comparada a abordagem anterior. Configurações com 100 clientes, 5 depósitos e 15 intervalos apresentam o maior tempo médio de execução, inferior a uma hora (3394s). Instâncias com 3 depósitos foram processadas, em média, após 866 segundos, enquanto aquelas com 5 depósitos exigiram um tempo médio de execução de 919 segundos.

Tabela 4.5: RESULTADOS MÉDIOS DA ABORDAGEM META-HEURÍSTICA

Tipo	$ V_c $	3 depósitos						5 depósitos					
		Melhor solução na etapa			Tempo total(s)	Melhoria sobre o modelo (%)		Melhor solução na etapa			Tempo total(s)	Melhoria sobre o modelo (%)	
		Inicial.	Evolução	Intensif.		Qualidade	Tempo	Inicial.	Evolução	Intensif.		Qualidade	Tempo
Pequena	10	27146,20	26998,80	26941,20	1	-0,14	88,78	26389,80	26281,00	26307,00	1	-0,01	99,59
	20	34492,80	33264,40	33150,80	36	0,32	99,66	32810,80	31942,40	31803,80	42	2,73	99,61
	50	55512,60	48361,00	48178,40	420	9,90	96,11	57520,60	48268,60	48066,20	341	16,29	96,85
	80	72538,60	63555,20	63116,20	1135	12,17	89,49	74424,00	61006,20	60435,00	990	14,75	90,83
	100	83422,60	74066,00	73297,20	2767	12,12	74,38	81921,00	66090,40	65456,40	2818	18,83	73,90
Média		54622,56	49249,08	48936,76	872	6,87	89,69	54613,24	46717,72	46413,68	838	10,52	92,16
Média	10	27030,60	26971,40	26844,20	1	-0,22	99,65	25520,20	25470,20	25272,00	1	-0,15	99,99
	20	33777,00	32980,80	32804,60	39	2,26	99,64	32303,60	31040,00	30884,60	61	4,90	99,43
	50	56894,40	49055,00	48631,20	362	10,85	96,65	54593,60	46949,40	46505,60	413	14,05	96,18
	80	73233,60	64505,00	63780,80	1178	12,30	89,09	72195,60	61483,20	60799,40	1119	15,11	89,64
	100	81764,00	72966,00	72227,80	3010	11,10	72,13	84361,20	66032,40	65237,00	3182	21,02	70,54
Média		54539,92	49295,64	48857,72	918	7,26	91,43	53794,84	46195,04	45739,72	955	10,99	91,15
Grande	10	27641,20	27630,40	27499,60	1	-0,11	99,92	26305,60	26305,60	26174,80	1	-0,01	99,93
	20	35377,40	33539,60	33299,60	42	2,34	99,61	32295,40	31588,80	31188,20	70	4,74	99,35
	50	56525,60	48973,00	48583,80	467	13,20	95,67	59277,60	47375,60	46865,40	408	19,78	96,22
	80	72522,00	63052,20	62227,60	943	13,14	91,27	76178,60	62939,00	62275,20	946	18,77	91,24
	100	82783,80	72826,40	71740,20	2588	12,46	76,04	81258,40	65826,40	65047,60	3394	18,98	68,57
Média		54970,00	49204,32	48670,16	808	8,21	92,50	55063,12	46807,08	46310,24	964	12,45	91,06
Média global		54710,83	49249,68	48821,55	866	7,45	91,21	54490,40	46573,28	46154,55	919	11,32	91,46

FONTE: A autora (2019)



Comparando esses resultados aos apresentados na Tabela 4.4, verifica-se uma melhoria média na qualidade das soluções de 7,45% para instâncias com 3 depósitos, e de 11,32% para instâncias com 5 depósitos. No tempo de execução, o ganho é consideravelmente maior, atingindo 91,21% e 91,46%, respectivamente, nas configurações com 3 e 5 potenciais depósitos. Isso destaca a relevância de tal abordagem, tanto na obtenção de soluções factíveis para o modelo exato, quanto na geração de soluções de qualidade, em um tempo computacional bastante inferior ao demandado pelo modelo de programação linear inteira mista proposto.

#### 4.4.3 Desempenho do algoritmo meta-heurístico na resolução do PLR-DT

Avalia-se aqui o desempenho da abordagem meta-heurística proposta na resolução do PLR-DT, definido na Seção 3.1. Para isso, algumas adaptações foram necessárias. Primeiro, eliminou-se da etapa de intensificação a heurística que permite atrasar o momento de saída dos veículos do depósito. Assim, todas as rotas iniciam no momento zero, conforme previsto na descrição do problema em questão. Além disso, considerou-se um único tipo de veículo (frota homogênea), com custo fixo igual a zero para o limite de veículos da frota, estabelecido nas instâncias originais do problema, e custo proporcional à capacidade para os veículos adicionais. Para todos os depósitos, a capacidade foi definida como a soma das demandas de todos os clientes, de forma que um único depósito pudesse atender a todos, e o custo de abertura diretamente proporcional à capacidade. Assim, considerou-se o mesmo custo para todos os depósitos, o qual foi descontado ao final da execução da meta-heurística, para coincidir com a proposta original do problema.

A Tabela 4.6 reporta os resultados médios alcançados com instâncias de 3 depósitos, e a Tabela 4.7 para os casos que contemplam 5 potenciais depósitos. Além disso, elas também mostram o desempenho dessa abordagem quando comparada ao algoritmo mateurístico proposto para o PLR-DT, na Seção 3.2. As duas primeiras colunas fornecem informações sobre as instâncias. Para facilitar a análise, na sequência, reprisam-se a melhor solução e o tempo de execução obtidos com a aplicação da mateurística da Seção 3.2. As duas colunas seguintes de cada tabela contêm os resultados obtidos com a meta-heurística. Da mesma forma, primeiro apresenta-se a melhor solução gerada e, na sequência, o tempo de execução. Por fim, as duas últimas colunas contemplam a melhoria proporcionada por esta última abordagem, em comparação à anterior, tanto no que tange a qualidade da solução quanto no tempo de processamento.

Os resultados da Tabela 4.6 mostram a superioridade da abordagem meta-heurística, adaptada para resolver o PLR-DT, em relação a mateurística proposta na Seção 3.2. Soluções médias de melhor qualidade foram produzidas para todas as instâncias, indiferente do número de clientes e de intervalos de tempo. A qualidade média dessas soluções foi 2,64% superior nas instâncias classificadas como pequenas, 2,38% nas médias e 2,07% nas grandes. Essa melhoria chega a 5,39% para configurações com 10 clientes e 5 intervalos. Resultados mais detalhados mostram ainda que a abordagem meta-heurística foi capaz de chegar a solução ótima em todas as instâncias com 10 clientes e 3 depósitos, independentemente do número de

Tabela 4.6: RESULTADOS MÉDIOS DA ABORDAGEM META-HEURÍSTICA NA RESOLUÇÃO DO PLR-DT PARA INSTÂNCIAS COM 3 DEPÓSITOS

Tipo	$ V_c $	Mateurística da Seção 3.2		Meta-heurística da Seção 4.2			
		Melhor	Tempo	Melhor	Tempo	Melhoria (%)	
		solução	(s)	solução	(s)	Qualidade	Tempo
Pequena	10	7436,00	38	7030,40	8	5,39	79,72
	20	13182,40	113	12860,80	64	2,40	47,20
	50	23624,20	739	22955,20	380	2,84	48,09
	80	33212,80	2710	32714,00	1083	1,53	54,80
	100	39759,80	5462	39267,40	2574	1,03	42,66
	Média	23443,04	1813	22965,56	821,81	2,64	54,49
Média	10	7413,40	36	7252,60	15	2,16	58,04
	20	13248,40	92	12742,80	49	3,86	46,15
	50	23451,60	883	22710,80	485	3,13	45,22
	80	34467,20	2512	34004,40	1188	1,36	51,58
	100	38704,20	4775	38170,20	1915	1,38	59,59
	Média	23456,96	1660	22976,16	730,47	2,38	52,11
Grande	10	7378,00	34	7153,00	32	3,02	1,85
	20	13328,80	89	12934,00	64	2,97	28,30
	50	23261,60	845	22631,40	503	2,75	37,87
	80	33240,80	2362	32983,80	815	0,80	64,54
	100	38691,20	4630	38391,20	2412	0,82	40,47
	Média	23180,08	1592	22818,68	765,14	2,07	34,60
	Média global	23360,03	1688	22920,13	772,47	2,36	47,07

FONTE: A autora (2019)

intervalos considerados. Além disso, o tempo de execução requerido foi significativamente menor, proporcionando uma redução média de 47,07% em relação à abordagem mateurística.

Da mesma forma, os resultados da Tabela 4.7 mostram que o algoritmo meta-heurístico produziu soluções médias de melhor qualidade para todas as configurações. Melhorias de 1,76%, 1,67% e 1,99% foram geradas, respectivamente, para as instâncias classificadas como pequena, média e grande. Um detalhamento desses resultados mostrou ainda que a meta-heurística foi capaz de produzir soluções ótimas em 14 das 15 instâncias com 10 clientes e que, além disso, na instância restante o resultado foi superior ao do algoritmo mateurístico proposto na Seção 3.2. Em relação ao tempo de execução, este também foi notavelmente menor, produzindo uma redução média de 58,39% quando comparado ao tempo requerido pela mateurística.

Isso mostra que o algoritmo meta-heurístico proposto para resolver o PLRDFH-DT pode facilmente ser adaptado para solucionar também o PLR-DT. Para todas as configurações testadas ele foi capaz de produzir soluções médias de melhor qualidade e exigiu um tempo de processamento substancialmente menor.

Tabela 4.7: RESULTADOS MÉDIOS DA ABORDAGEM META-HEURÍSTICA NA RESOLUÇÃO DO PLR-DT PARA INSTÂNCIAS COM 5 DEPÓSITOS

Tipo	$ V_c $	Mateurística da Seção 3.2		Meta-heurística da Seção 4.2			
		Melhor	Tempo	Melhor	Tempo	Melhoria (%)	
		solução	(s)	solução	(s)	Qualidade	Tempo
Pequena	10	7767,80	64	7604,00	63	2,07	-8,14
	20	12073,20	200	11771,80	73	2,58	62,47
	50	23107,40	1354	22700,20	794	1,71	41,11
	80	30723,00	4169	30213,60	897	1,53	78,38
	100	33290,60	9933	33031,80	2195	0,93	77,60
Média		21392,40	3144	21064,28	804	1,76	50,28
Média	10	7833,00	56	7788,40	14	0,75	74,65
	20	12609,20	160	12176,20	64	3,45	60,56
	50	22985,40	1543	22598,80	621	1,62	61,38
	80	30604,20	3082	29890,00	1035	2,33	66,43
	100	33065,00	9365	32892,80	2434	0,17	74,10
Média		21419,36	2841	21069,24	834	1,67	67,43
Grande	10	7479,80	56	7355,40	26	1,61	53,10
	20	12376,60	157	11954,80	76	3,45	50,73
	50	23406,00	1504	22923,60	849	2,04	43,66
	80	31151,60	3820	30408,40	1125	2,34	66,67
	100	33185,20	9613	33037,00	2530	0,51	73,14
Média		21519,84	3030	21135,84	921	1,99	57,46
Média global		21443,87	3005,10	21089,79	853	1,81	58,39

FONTE: A autora (2019)

## 4.5 CONCLUSÕES

Este capítulo contempla a definição de uma extensão do problema de localização e roteamento dependente do tempo, que inclui dimensionamento de frota heterogênea, e uma formulação baseada em programação linear inteira mista. Ela contribui com a área de logística integrada, uma vez que apresenta um problema ainda não tratado na literatura. O modelo matemático proposto é capaz de fornecer soluções para instâncias com até 5 depósitos, 100 clientes e 15 intervalos de tempo, que consideram a variabilidade nas condições de tráfego ao longo do dia. O provimento de soluções iniciais e a inclusão de um conjunto de desigualdades válidas são de extrema importância para a melhoria dos limites inferiores. Para elevar a qualidade das soluções e reduzir o tempo computacional, propôs-se ainda um algoritmo meta-heurístico de resolução. Seu desempenho foi comparado com a formulação exata, em um conjunto de instâncias baseado em dados reais de tráfego. Mostrou-se a importância de uma abordagem que integre adequadamente diferentes heurísticas para tratar problemas complexos como este. O algoritmo proposto foi capaz de gerar soluções médias de melhor qualidade em todas as configurações de instância com mais de 10 clientes, além de reduzir significativamente o tempo computacional.

## 5 CONSIDERAÇÕES FINAIS

A integração de decisões e o esforço para encontrar soluções eficientes para problemas práticos tornaram-se questões de interesse crescente dentro da área de logística integrada. Neste trabalho, buscou-se preencher uma lacuna existente entre dois tópicos populares de pesquisa: o problema de localização e roteamento (PLR) e o problema de roteamento de veículos dependente do tempo (PRV-DT). Decisões de caráter estratégico e operacional foram integradas e inseridas em um cenário que contempla as variações na velocidade dos veículos ao longo do dia, provocadas por diferentes condições de tráfego. Foram propostos, modelados e resolvidos dois problemas de localização e roteamento de veículos dependentes do tempo. O primeiro, estruturado a partir de características que definem um PLR clássico; e o segundo, uma extensão que visa aproximar a definição do problema das condições reais, através de inclusão de atributos adicionais (frota heterogênea e não limitada, depósitos capacitados, flexibilidade no horário de saída dos veículos do depósito).

No Capítulo 2, depois de discorrer sobre o problema de localização e roteamento, abordou-se a dependência de tempo em problemas que envolvem roteirização de veículos. Primeiro, focou-se na versão clássica e nas principais características que definem o PLR. Abordaram-se as variantes e extensões mais relevantes, os modelos de resolução exata, limites inferiores e abordagens heurísticas recentes. Na sequência, apresentaram-se problemas de roteamento que contemplam o aspecto temporal relacionado ao deslocamento dos veículos, já estudados na literatura. A revisão mostrou que, apesar de recente, o aspecto da dependência de tempo já foi amplamente abordado em problemas de roteirização, mas constitui uma lacuna no que tange à logística integrada.

No Capítulo 3, introduziu-se o problema de localização e roteamento dependente do tempo. Decisões táticas, associadas à utilização dos depósitos, e operacionais, ligadas à execução de rotas, foram otimizadas simultaneamente. Elas estão inseridas em um contexto de frota limitada e homogênea, onde o horário de saída dos veículos nos depósitos é pré-estabelecido. Gerou-se um conjunto de instâncias baseadas em dados reais de tráfego. Propôs-se uma formulação matemática, reforçada por um conjunto de desigualdades válidas. Soluções iniciais foram geradas a partir de heurísticas construtivas. Estas, juntamente com um modelo de particionamento de conjunto, integram um algoritmo metaheurístico. Intensivos testes computacionais foram realizados, e os resultados revelaram a importância da integração de métodos heurísticos e modelos exatos.

Uma extensão do problema foi proposta no Capítulo 4. Incluiu-se limitação de capacidade nos depósitos e custos fixos de abertura, além de frota heterogênea e não limitada, com custos fixos de utilização por veículo. Adicionou-se ainda a flexibilidade ao horário de saída dos veículos nos depósitos. Assim, decisões de localização foram integradas às questões de roteamento e dimensionamento da frota. O problema foi resolvido a partir de um modelo de programação

linear inteira mista, e fortalecido por um conjunto desigualdades válidas. Na sequência, para superar a limitação do método exato na resolução de instâncias maiores, propôs-se um algoritmo meta-heurístico baseado em busca evolucionária. Extensivos testes computacionais foram conduzidos para diferentes configurações de instância. Mostrando a limitação da formulação matemática, destacou-se a importância de desenvolver bons algoritmos heurísticos de resolução.

Esta tese contribui com o desenvolvimento da pesquisa na área de logística integrada de várias formas. Primeiro, por introduzir e tratar de problemas ainda não abordados na literatura. Ao incorporar a dependência temporal às decisões de localização e roteamento, contempla-se uma simplificação até então presente nos trabalhos da área: assumir que a velocidade é constante ao longo do dia e invariável, independentemente da região ou via considerada. Segundo, por propor a primeira formulação matemática da literatura para essa classe de problemas. Este modelo geral serviu de base para a inclusão de novas características e pode auxiliar na modelagem de casos reais, no desenvolvimento de outras extensões ou na criação de novas variantes do problema. Terceiro, pelo desenvolvimento de duas abordagens heurísticas de resolução, moldadas para contemplar a dependência de tempo. Ambas são capazes de tratar instâncias com até 100 clientes, 5 depósitos e 15 intervalos de tempo. De modo adicional, pode-se citar ainda a geração do primeiro conjunto de instâncias de teste para o PLR-DT, baseado em dados reais de tráfego. Por fim, minimizar o tempo de deslocamento dos veículos também significa, indiretamente, localizar instalações e traçar rotas de modo a evitar regiões e horários de tráfego mais intenso. Isso reflete, de forma direta, na redução das emissões de gases de efeito estufa. Assim, este trabalho também contribui com os avanços nesse sentido.

Resolver um problema integrado representa uma tarefa complexa. Para que o modelo seja tratável e as soluções possam ser implementadas é necessário fazer algumas simplificações e delimitar a pesquisa. Aqui, limitou-se o estudo ao considerar, por exemplo, apenas demandas e tempos de atendimento determinísticos; e assumir velocidade de deslocamento constante dentro de um intervalo de tempo, e invariável, mesmo quando a travessia do arco inicia e finaliza em intervalos distintos. Nesta tese, utilizaram-se instâncias criadas aleatoriamente, com base em dados reais de tráfego. Apesar do caráter aleatório presente nessa geração, os resultados dos testes experimentais são válidos para o conjunto em questão. Finalmente, a abordagem adotada nesta pesquisa contemplou, primeiramente, a definição dos problemas e sua resolução através de métodos exatos. Posteriormente, a partir da necessidade, desenvolveram-se e aplicaram-se métodos heurísticos de solução. Em ambos os casos, a complexidade dos problemas torna sua otimização difícil à medida que aumenta o tamanho das instâncias e o número de variáveis.

Posto isso, ainda existem oportunidades de ampliar a pesquisa em trabalhos futuros. Sugerem-se inovações do ponto de vista prático e também em relação ao método de resolução. No que tange à aplicação, outros elementos podem ser levados em consideração no problema, tais como janelas de tempo no atendimento dos clientes. Além disso, para casos práticos onde nem sempre é possível atender a todas as demandas, mas a manutenção do nível de atendimento dos clientes é o objetivo principal, sugere-se explorar a taxa de *backlog*. Do ponto

de vista do método de resolução, recomenda-se avaliar a possibilidade de aplicar a decomposição baseada em cortes combinatorias de Benders como estratégia na resolução do modelo. Investigar uma abordagem temporal com discretização não uniforme também é uma proposta, além de implementar a abordagem de solução executando ainda a formulação matemática dentro de um conceito de horizonte rolante. Ademais, sugere-se estudar a possibilidade dos algoritmos se tornarem adaptativos. Isso pode ser avaliado tanto em relação às heurísticas ou operadores usados, quanto no que diz respeito aos parâmetros, que podem evoluir ao longo das gerações.

Outras oportunidades de pesquisa neste campo se encontram na investigação de problemas estocásticos e dinâmicos, e na incorporação de aspectos relacionados a outros níveis da cadeia de suprimentos. Além do mais, também pode-se considerar o desenvolvimento de métodos exatos mais rápidos e eficientes e metodologias capazes de tratar grandes instâncias. Uma extensão interessante desta tese seria integrar as decisões de localização, roteamento e controle de estoque, considerando a dependência temporal.

## REFERÊNCIAS

- AKYOL, D.; KOSTER, R. D. Determining time windows in urban freight transport: A city cooperative approach. *Transportation Research Part E: Logistics and Transportation Review*, v. 118, p. 34–50, 2018.
- ALBAREDA SAMBOLA, M.; DIAZ, J.; FERNANDEZ, E. A compact model and tight bounds for a combined location-routing problem. *Computers & Operations Research*, v. 32, n. 3, p. 407–428, Mar. 2005.
- ALVAREZ, A.; MUNARI, P.; MORABITO, R. Iterated local search and simulated annealing algorithms for the inventory routing problem. *International Transaction in Operational Research*, v. 25, p. 1785–1809, 2018.
- ALVAREZ, P. et al. The impact of traffic congestion when optimising delivery routes in real time. A case study in Spain. *International Journal of Logistics Research and Applications*, Taylor & Francis, v. 21, n. 5, p. 529–541, 2018.
- AMBROSINO, D.; SCIOMACHEN, A.; SCUTELLÀ, M. A heuristic based on multiexchange techniques for a regional fleet assignment location-routing problem. *Computers & Operations Research*, v. 36, p. 442–460, 2009.
- BALDACCI, R.; MINGOZZI, A.; WOLFLER CALVO, R. An exact method for the capacitated location-routing problem. *Operations Research*, v. 59, n. 5, p. 1284–1296, Out. 2011.
- BARRETO, S. *Análise e modelização de problemas de localização-distribuição*. 337 f. Tese (Doutorado em Gestão Industrial) — Universidade de Aveiro, Aveiro, Portugal, 2004.
- BEHNKE, M.; KIRSCHSTEIN, T. The impact of path selection on GHG emissions in city logistics. *Transportation Research Part E: Logistics and Transportation Review*, v. 106, p. 320–336, 2017.
- BEKTAŞ, T.; CRAINIC, T.; WOENSEL, T. V. From managing urban freight to smart city logistics networks. In: GAKIS, K.; PARDALOS, P. (Ed.). *Network Design and Optimization for Smart Cities*. [S.l.]: World Scientific, 2017. cap. 7, p. 143–188.
- BELENGUER, J.-M. et al. A branch-and-cut method for the capacitated location-routing problem. *Computers & Operations Research*, v. 38, n. 6, p. 931–941, Jun. 2011.
- BELHASSINE, K. et al. *Improved home deliveries in congested areas using geospatial technology*. Working Paper CIRRELT-2018-02, 2018.
- BERTAZZI, L. et al. A matheuristic algorithm for the multi-depot inventory routing problem. *Transportation Research Part E*, v. 122, p. 524–544, 2019.
- BOWERSOX, D.; CALANTONE, R.; RODRIGUES, A. Estimation of global logistics expenditures using neural networks. *Journal of Business Logistics*, Elsevier, v. 24, n. 2, p. 21–36, 2003.
- BREUNIG, U. et al. A large neighbourhood based heuristic for two-echelon routing problems. *Computers & Operations Research*, v. 76, p. 208–225, 2016.

- CAPELLE, T. et al. A column generation approach for location-routing problems with pickup and delivery. *European Journal of Operational Research*, v. 272, p. 121–131, 2019.
- CLARKE, G.; WRIGHT, J. Scheduling of vehicles from a central depot to a number of delivery points. *Operations Research*, v. 12, p. 568–581, 1964.
- CLEOPHAS, C. et al. Collaborative urban transportation: Recent advances in theory and practice. *European Journal of Operational Research*, v. 273, p. 801–816, 2019.
- CONTARDO, C.; CORDEAU, J.; GENDRON, B. An exact algorithm based on cut-and-column generation for the capacitated location-routing problem. *INFORMS Journal on Computing*, v. 26, n. 1, p. 88–102, Jun. 2014.
- CONTARDO, C.; CORDEAU, J.-F.; GENDRON, B. A GRASP+ ILP-based metaheuristic for the capacitated location-routing problem. *Journal of Heuristics*, v. 20, n. 1, p. 1–38, Feb. 2014.
- CORDEAU, J.-F.; GHIANI, G.; GUERRIERO, E. Analysis and branch-and-cut algorithm for the time-dependent travelling salesman problem. *Transportation Science*, v. 48, n. 1, p. 46–58, Feb. 2014.
- DAI, Z. et al. A two-phase method for multi-echelon location-routing problems in supply chains. *Expert Systems with Applications*, v. 115, p. 618–634, 2019.
- DARVISH, M. et al. Flexible two-echelon location routing problem. *European Journal of Operational Research*, v. 277, p. 1124–1136, 2019.
- DELLING, D.; NANNICINI, G. Core routing on dynamic time-dependent road networks. *INFORMS Journal on Computing*, v. 24, n. 2, p. 187–201, 2012.
- DEMIR, E.; BEKTAŞ, T.; LAPORTE, G. An adaptative large neighborhood search heuristic for the pollution-routing problem. *European Journal of Operational Research*, v. 223, p. 346–359, 2012.
- DEMIR, E.; BEKTAŞ, T.; LAPORTE, G. A review of recent research on green road freight transportation. *European Journal of Operational Research*, Elsevier, v. 237, n. 3, p. 775–793, 2014.
- DOULABI, S. H.; SEIFI, A. Lower and upper bounds for location-arc routing problems with vehicle capacity constraints. *European Journal of Operational Research*, v. 224, p. 189–208, 2013.
- DREXL, M.; SCHNEIDER, M. A survey of variants and extensions of the location-routing problem. *European Journal of Operational Research*, v. 241, n. 2, p. 283–308, 2015.
- DREXL, M.; SCHNEIDER, M. A survey of the standard location-routing problem. *Annals of Operations Research*, v. 259, n. 1-2, p. 389–414, 2017.
- DUHAMEL, C. et al. A GRASPxELS approach for the capacitated location routing problem. *Computers & Operations Research*, v. 37, n. 11, p. 1912–1923, 2010.
- DUKKANCI, O.; KARA, B.; BEKTAŞ, T. The green location-routing problem. *Computers and Operations Research*, Elsevier, v. 105, p. 187–202, 2019.



- EHMKE, J.; CAMPBELL, A.; THOMAS, B. Data-driven approaches for emissions-minimized paths in urban areas. *Computers & Operations Research*, v. 67, p. 34–47, 2016.
- ESCOBAR, J. et al. A granular variable tabu neighborhood search for the capacitated location-routing problem. *Transportation Research Part B: Methodological*, v. 67, p. 344–356, Set. 2014.
- ESCOBAR, J.; LINFATI, R.; TOTH, P. A two-phase hybrid heuristic algorithm for the capacitated location-routing problem. *Computers & Operations Research*, v. 40, n. 1, p. 70–79, Jan. 2013.
- GAO, S. et al. Ant colony optimization with clustering for solving the dynamic location routing problem. *Applied Mathematics and Computation*, v. 285, p. 149–173, 2016.
- GENDREAU, M.; GHIANI, G.; GUERRIERO, E. Time-dependent routing problems: a review. *Computers & Operations Research*, v. 64, p. 189–197, 2015.
- GOLDBERG, D.; LINGLE, J. Alleles, loci, and the traveling salesman problem. In: *Proceedings of the First International Conference on Genetic Algorithms and Their Applications*. Pittsburg - United States: [s.n.], 1985. p. 154–159.
- GOLDEN, B. et al. The fleet size and mix vehicle routing problem. *Computers & Operations Research*, v. 11, p. 49–66, 1984.
- HEMMELMAYR, V.; CORDEAU, J.-F.; CRAINIC, T. An adaptive large neighborhood search heuristic for the two-echelon vehicle routing problems arising in city logistics. *Computers & Operations Research*, v. 39, p. 3215–3228, 2012.
- HU, L. et al. Joint design of fleet size, hub locations, and hub capacities for third-party logistics networks with road congestion constraints. *Transportation Research Part E: Logistics and Transportation Review*, v. 118, p. 568–588, 2018.
- ICHOUA, S.; GENDREAU, M.; POTVIN, J.-Y. Vehicle dispatching with time-dependent travel times. *European Journal of Operational Research*, Elsevier, v. 144, n. 2, p. 379–396, 2003.
- JABALLAH, R. et al. *The time-dependent shortest path and vehicle routing problem*. Working Paper CIRRELT-2019-12, 2019.
- KARA, I.; LAPORTE, G.; BEKTAŞ, T. A note on the lifted Miller-Tucker-Zemlin subtour elimination constraints for the capacitated vehicle routing problem. *European Journal of Operational Research*, Elsevier, v. 158, n. 3, p. 793–795, 2004.
- KOÇ, Ç. A unified-adaptative large neighborhood search metaheuristic for periodic location-routing problems. *Transportation Research Part C*, v. 68, p. 265–284, 2016.
- KOÇ, Ç. et al. The fleet size and mix location-routing problem with time windows: formulations and a heuristic algorithm. *European Journal of Operational Research*, v. 248, n. 1, p. 33–53, Jan. 2016.
- KOÇ, Ç. et al. A hybrid evolutionary algorithm for heterogeneous fleet vehicle routing problems with time windows. *Computers & Operations Research*, v. 64, p. 11–27, 2015.
- KOÇ, Ç. et al. The impact of depot location, fleet composition and routing on emissions in city logistics. *Transportation Research Part B: Methodological*, Elsevier, v. 84, p. 81–102, 2016.

- KULKARNI, R.; BHAVE, P. Integer programming formulations of vehicle routing problems. *European Journal of Operational Research*, v. 20, n. 1, p. 58–67, 1985.
- LAHYANI, R.; COELHO, L.; RENAUD, J. Alternative formulations and improved bound for the multi-depot fleet size and mix vehicle routing problem. *OR Spectrum*, v. 40, p. 125–157, 2018.
- LAPORTE, G.; NOBERT, Y. An exact algorithm for minimizing routing and operating costs in depot location. *European Journal of Operational Research*, v. 6, p. 224–226, 1981.
- LOPES, R. et al. A taxonomical analysis, current methods and objectives on location-routing problems. *International Transactions in Operational Research*, v. 20, p. 795–822, 2013.
- LUAN, S. et al. The HSABA for emergency location-routing problem. *Mathematical Problems in Engineering*, v. 2019, p. 12 pages, 2019.
- MALANDRAKI, C.; DASKIN, M. Time dependent vehicle routing problems: formulations, properties and heuristic algorithms. *Transportation Science*, INFORMS, v. 26, n. 3, p. 185–200, 1992.
- MANCINI, S. A combined multistart random constructive heuristic and set partitioning based for mulation for the vehicle routing problem with time dependent travel times. *Computers & Operations Research*, v. 88, p. 290–296, 2017.
- MANZOUR AL AJDAD, S.; TORABI, S.; SALHI, S. A hierarchical algorithm for the planar single-facility location routing problem. *Computers & Operations Research*, v. 39, n. 2, p. 461–470, Feb. 2012.
- MARANZANA, F. On the location of supply points to minimize transport costs. *Operational Research Quarterly*, v. 15, n. 3, p. 261–270, 1964.
- MARINAKIS, Y.; MARINAKI, M.; MIGDALAS, A. A hybrid clonal algorithm for the location routing problem with stochastic demands. *Annals of Mathematics and Artificial Intelligence*, v. 76, p. 121–142, 2016.
- MARUFUZZAMAN, M.; EKŞİOĞLU, S. D. Managing congestion in supply chains via dynamic freight routing: An application in the biomass supply chain. *Transportation Research Part E: Logistics and Transportation Review*, v. 99, p. 54–76, 2017.
- MILLER-HOOKS, E.; YANG, B. Updating paths in time-varying networks given arc weight changes. *Transportation Science*, v. 39, n. 4, p. 451–464, 2005.
- MIN, H.; JAYARAMAN, V.; SRIVASTAVA, R. Combined location-routing problems: A synthesis and future research directions. *European Journal of Operational Research*, Elsevier, v. 108, n. 1, p. 1–15, 1998.
- NADIZADEH, A.; KAFASH, B. Fuzzy capacitated location-routing problem with simultaneous pickup and delivery demands. *Transportation Letters*, v. 11, n. 1, p. 1–19, 2019.
- NAGY, G.; SALHI, S. Location-routing: issues, models and methods. *European Journal of Operational Research*, v. 177, n. 2, p. 649–672, 2007.
- NATARAJ, S. et al. Consolidation centers in city logistics: a cooperative approach based on the location routing problem. *International Journal of Industrial Engineering Computations*, v. 10, p. 393–404, 2019.

- PEKEL, E.; KARA, S. Solving capacitated location routing problem by variable neighborhood descent and GA-artificial neural network hybrid method. *Promet*, v. 30, n. 5, p. 563–578, 2018.
- PRINS, C. et al. Solving the capacitated location-routing problem by a cooperative lagrangean relaxation-granular tabu search heuristic. *Transportation Science*, v. 41, n. 4, p. 470–483, Nov. 2007.
- PRODHON, C.; PRINS, C. A survey of recent research on location-routing problems. *European Journal of Operational Research*, v. 238, n. 1, p. 1–17, Out. 2014.
- RABBANI, M.; DAVOUDKHANI, M.; FARROKHI-ASL, H. A new multi-objective green location routing problem with heterogeneous fleet of vehicles and fuel constraint. *International Journal of Strategic Decision Sciences*, v. 8, n. 3, p. 99–119, 2017.
- ROPKE, S.; PISINGER, D. An adaptative large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, v. 40, p. 455–472, 2006.
- SALHI, S.; NAGY, G. Consistency and robustness in location-routing. *Studies in Locational Analysis*, n. 13, p. 3–19, 1999.
- SALHI, S.; RAND, G. The effect of ignoring routes when locating depots. *European Journal of Operational Research*, v. 39, n. 2, p. 150–156, 1989.
- SCHIFFER, M.; WALTHER, G. The electric location routing problem with time windows and partial recharging. *European Journal of Operational Research*, v. 260, p. 995–1013, 2017.
- SCHMIDT, C. E. et al. The time-dependent location-routing problem. *Transportation Research Part E*, v. 128, p. 293–315, 2019.
- SCHNEIDER, M.; LÖFFLER, M. Large composite neighborhoods for the capacitated location-routing problem. *Transportation Science*, v. 53, n. 1, p. 301–318, 2019.
- SUNG, K. et al. Shortest paths in a network with time-dependent flow speeds. *European Journal of Operational Research*, v. 121, n. 1, p. 32–39, 2000.
- TING, C.-J.; CHEN, C.-H. A multiple ant colony optimization algorithm for the capacitated location routing problem. *International Journal of Production Economics*, v. 141, n. 1, p. 34–44, Jan. 2013.
- TORO, E. et al. A multi-objective model for the green capacitated location-routing problem considering environmental impact. *Computers & Industrial Engineering*, v. 110, p. 114–125, 2017.
- VEENSTRA, M.; COELHO, L. *The time-dependent shortest path and vehicle routing problem*. Working Paper FSA-2017-012. Faculté des sciences de l'administration de l'Université Laval, 2017.
- VEENSTRA, M. et al. A simultaneous facility location and vehicle routing problem arising in health care logistic in the Netherlands. *European Journal of Operational Research*, v. 268, p. 703–715, 2018.
- VON BOVENTER, E. The relationship between transportation costs and location rent in transportation problems. *Journal of Regional Science*, v. 3, n. 2, p. 27–40, 1961.

WANG, J. et al. Multiobjective multiple neighborhood search algorithms for multiobjective fleet size and mix location-routing problem with time windows. *IEEE Transactions on Systems, man, and cybernetics systems*, v. 99, p. 1–15, 2019.

WANG, X.; YANG, F.; LU, D. Multi-objective location-routing problem ith simultaneous pickup and delivery for urban distribution. *Journal of Intelligent & Fuzzy Systems*, v. 35, p. 3987–4000, 2018.

WATSON GANDY, C.; DOHRN, P. Depot location with van salesmen - a practical approach. *Omega*, v. 1, n. 3, p. 321–329, Jun. 1973.

YU, V. et al. A simulated annealing heuristic for the capacitated location routing problem. *Computers & Industrial Engineering*, v. 58, n. 2, p. 288–299, Mar. 2010.

YUAN, Y.; YU, J. Locating transit hubs in a multi-modal transportation network: A cluster-based optimization approach. *Transportation Research Part E: Logistics and Transportation Review*, Elsevier, v. 114, p. 85–103, 2018.

ZHAO, Q.; EANG, W.; SOUZA, R. D. A heterogeneous fleet two-echelon capacitated location-routing model for joint delivery arising in city logistics. *International Journal of Production Research*, v. 56, n. 15, p. 5062–5080, 2018.

## 6 APÊNDICE A

A formulação matemática proposta para a abordagem sequencial do PRV-DT é descrita na sequência, onde  $V_d$  representa o conjunto unitário que contém o depósito de índice 0:

$$\min \sum_{(i,j) \in A} \sum_{h \in H} t_{ij}^h x_{ij}^h \quad (6.1)$$

sujeito à:

$$\sum_{i \in (V_c \setminus \{j\}) \cup V_d} z_{ij} = 1, \quad \forall j \in V_c \quad (6.2)$$

$$\sum_{j \in (V_c \setminus \{i\}) \cup V_t} z_{ij} = 1, \quad \forall i \in V_c \quad (6.3)$$

$$\sum_{j \in V_c} z_{0j} \leq |K| \quad (6.4)$$

$$\sum_{i \in V_c} z_{ij} \leq 1, \quad \forall j \in V_t \quad (6.5)$$

$$u_i - u_j + Q z_{ij} \leq Q - q_j, \quad \forall i, j \in V_c, i \neq j \quad (6.6)$$

$$q_i \leq u_i \leq Q, \quad \forall i \in V_c \quad (6.7)$$

$$\sum_{i \in V_c} x_{0i}^0 = \sum_{i \in V_c} \sum_{j \in V_t} z_{ij} \quad (6.8)$$

$$\sum_{h \in H} x_{ij}^h = z_{ij}, \quad \forall (i, j) \in A \quad (6.9)$$

$$x_{ij}^h \leq y_i^h, \quad \forall (i, j) \in A, \forall h \in H \quad (6.10)$$

$$\sum_{h \in H} y_i^h = 1, \quad \forall i \in V \setminus V_t \quad (6.11)$$

$$a_0 = 0 \quad (6.12)$$

$$a_j \geq a_i + s_j + t_{ij}^h - 2\bar{T}(m+1)(1 - x_{ij}^h), \quad \forall (i, j) \in A, \forall h \in H \quad (6.13)$$

$$a_j \leq a_i + s_j + t_{ij}^h + \bar{T}(m+1)(1 - x_{ij}^h), \quad \forall (i, j) \in A, \forall h \in H \quad (6.14)$$

$$\sum_{h \in H} h\bar{T}y_i^h \leq a_i \leq \sum_{h \in H} h\bar{T}y_i^h + \bar{T} - \varepsilon, \quad i \in V_c \quad (6.15)$$

$$a_j \leq \sum_{i \in V_c} z_{ij}[\bar{T}(m+1) - \varepsilon], \quad \forall j \in V_t \quad (6.16)$$

$$z_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A \quad (6.17)$$

$$y_i^h \in \{0, 1\}, \quad \forall i \in V \setminus V_t, \forall h \in H \quad (6.18)$$

$$x_{ij}^h \in \{0, 1\}, \quad \forall (i, j) \in A, \forall h \in H \quad (6.19)$$

$$a_i \in \mathbb{R}_+ \quad \forall i \in V \quad (6.20)$$

$$u_i \in \mathbb{R}_+, \quad \forall i \in V_c. \quad (6.21)$$